



Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

Facultat de Matemàtiques
Universitat de Barcelona

Aplicació mòbil offline d'un gestor
de comandes genèric

Ricard Espinàs Llovet

Directora: Anna Puig Puig
Realitzat a: Departament de Matemàtica
Aplicada i Anàlisi. UB

Barcelona, 20 de juny de 2014

Índex

1. Introducció	4
1.1 Summary of the TFG	4
1.2 Projecte	5
1.3 Motivació	6
1.4 Objectius	7
2. Anàlisi del tipus de desenvolupament	9
2.1 Elecció del tipus d'aplicació mòbil	9
2.2 Aplicació Nativa o Híbrida?	11
2.3 Investigació de Frameworks Híbrid	13
2.4 Frameworks Híbrids més importants	17
2.4.1 Phonegap	17
2.4.2 RhoMobile	18
2.4.3 Titanium	19
2.5 Elecció Final del Framework Híbrid	20
3. Anàlisi de l'aplicació	22
3.1 Anàlisi de compradors interessats	22
3.2 Anàlisi d'usuaris	22
3.3 Anàlisi funcional	22
3.3.1 Tipus de Vistes	23
3.3.1.1 Vistes d'usuari no identificat	23
3.3.1.2 Vistes d'usuari identificat	25
3.3.1.3 Vistes de venedor	26
4. Casos d'Ús	28
4.1 Registrar Usuari	28
4.2 Identificar Usuari	29
4.3 Recordar Contrasenya	29
4.4 Localitzar Producte	30
4.5 Ordenar Llista de Productes Actual	30
4.6 Cercar Per Nom	31
4.7 Filtrar Productes	31
4.8 Afegir Producte al Carro de la Compra	32
4.9 Preparar comanda	33
4.10 Realitzar una comanda	33
4.11 Localitzar l'historial de comandes	34
4.12 Crear Llista Preferida del Carro de la Compra	34
4.13 Localitzar Llista Preferida	35
4.14 Elaborar el Carro de la compra a partir dels productes d'una Comanda o Llista Preferida	35
4.15 Eliminar llista preferida	36
4.16 Canviar d'estat una comanda	36
5. Disseny	38
5.1 Explicació del disseny	38
5.2 Millores extretes de simulacions i proves	41
5.2.1 Prova cas d'ús "Identificar d'usuari"	42
5.2.2 Prova cas d'ús "Afegir producte, preparar i realitzar Comanda"	43
5.2.3 Proves de l'aplicació en diferents negocis	44

6. Conclusions i possibles ampliacions.....	46
6.1 Conclusions	46
6.2 Possibles ampliacions o variacions	48
7. Referències.....	49
8. Annex I: Manual Tècnic.....	51
8.1 Requeriments i instal·lació de la aplicació per a un usuari.....	51
8.2 Requeriments i instal·lació de l'entorn de desenvolupament del projecte.....	51
8.2.1 Instal·lació de Node.js.....	51
8.2.2 Instal·lació Apache ant	52
8.2.3 Instal·lació de Java Standard Edition (Java SE).....	53
8.2.4 Instal·lació de Phonegap	54
8.2.5 Instal·lació d'Eclipse amb SDK's d'Android	54
8.2.6 Importació del projecte Phonegap	55
8.2.7 Testejar el projecte TFG Phonegap al IDE Eclipse.....	56
8.2.7.1 Simular mitjançant AVD	56
8.2.7.2 Executar mitjançant un dispositiu mòbil Android.....	57
8.2.8 Compilar i Exportar amb Phonegap Build.....	58
8.3 Requeriments i instal·lació del servidor del projecte	61

1. Introducció

1.1 Summary of the TFG

In this TFG we propose on a great product that might completely change the current market, mainly for the small or medium companies and for our own use, because the cost of this service it's far way cheaper than the others development systems.

We are talking about an application that is compatible with almost all mobile devices. To make it possible, the app will be compatible with almost all mobile operative systems and the different screen resolutions that is going to be used by the devices.

This app consists in an on-line sales management that works with generic products list. Like all this kind of applications, it will require a server but this app that I am exposing will work offline almost all the time, just when you log in or when you create/change some user information like a sale you must be online, and you will be connected to an external server ready for the application.

This Online/Offline sales management will have two kind of users: the customer and the seller. The customer is allowed to make requests, create favourite lists and he is able to check in different ways all the company products. In the other hand, the seller will have acces to an area where you can check all customer details, and additioally, they will have access too to an area to look for new customers requests and new customers that want to join this application. The seller will have an acces that a normal customer does not.

To manage the possible consults or modifications of the information we will use an extern data base associated with a server.

The application can be used with any kind of products or services. The easy interface used it's compatible for all ages.

1.2 Projecte

En aquest TFG es dissenya un nou producte que pot tenir molt bona rebuda al mercat actual, sobretot per a petites i mitjanes empreses i per a particulars ja que el cost de final d'aquest producte és un cost molt més baix tant en la producció com en el manteniment respecte a altres maneres de desenvolupament.

Es tracte d'una aplicació mòbil compatible amb el màxim de mòbils possibles, contemplant el màxim número de sistemes operatius mòbils existents, i també tenint en compte les possibles resolucions que aquests dispositius tenen.

Està enfocat per a un gestionar de ventes online d'una llista genèrica de productes, sigui quina sigui. Malgrat que no pugui deixar de dependre d'un servidor en la seva totalitat, la aplicació funcionarà a nivell offline per a la majoria d'accions que desenvoluparà, excepte casos necessaris com la identificació d'usuari, la càrrega d'informació pròpia de l'usuari, etc. en que es connectarà a un servidor extern preparat per a la aplicació.

Disposarà de dos tipus d'usuari: el client i el venedor. El client normal podrà realitzar comandes, elaborar llistes preferides de productes i visualitzar de diferents maneres els productes de l'empresa. La diferència entre el client i el venedor és que el venedor com el seu nom indica es dedicarà a buscar noves comandes de clients existents o possibles nous clients. Podrà consultar dades que un client normal no podrà accedir, etc.

Per a gestionar possibles consultes o modificacions de les dades utilitzarem una base de dades externa associada a un servidor.

La aplicació està pensat per a poder ser utilitzat per a tot games de productes o serveis que es comportin com a tal i la interfície pensada per a un fàcil ús és apte per a usuaris de totes les edats.

1.3 Motivació

Les aplicacions de mòbil d'aquest tipus sempre acostumen a necessitar Internet per a la càrrega d'informació, imatges i segons quines funcions, com ara la identificació d'usuari o accedir a les llistes de productes preferides o comandes antigues de l'usuari, en aquest cas.

A priori, sembla una solució òptima, ja que la aplicació que es descarrega de les App Store ocupa poc espai, té una ràpida descàrrega, etc.

Però poca gent ha pensat en què passa sinó disposem d'Internet o tenim poca cobertura. Quants cops ens hem queixat de que va lent l'Internet i hem acabat per aplaçar la tasca que volíem desenvolupar perquè ens molestava que es mostrés el missatge de carregant? Si és una aplicació que hem d'utilitzar repetidament, pot causar moltes molèsties.

En el passat, la memòria del telèfon era un punt important a tenir en compte per als requisits d'una aplicació. Com que es disposava de poc espai, els programadors feien les aplicacions molt petites per a que no reduís dràsticament l'espai lliure del telèfon i també s'havia de fer aplicacions tenint en compte els càlculs intensius i efectes visuals, ja que si s'exigia molt era probable que el mòbil no pogués processar fluidament les funcions demanades.

Avui en dia, disposem d'una nova generació de mòbils amb un processament i un emmagatzematge casi equiparables al d'un ordinador de sobretaula. Tenir un mòbil amb un processador Quad Core amb 1 Gb de memòria Ram i 8 Gbs de memòria disc dur ja no és ninguna bogeria.

Així doncs, tenint aquest nou context, podem pensar de fer just el que s'evitava en el passat. Tota la informació que necessita la aplicació pot estar dins de la aplicació instal·lada al mòbil i deixar per a l'Internet només les tasques que siguin completament necessàries com la identificació d'un usuari, la petició de comandes de productes, etc.

D'aquesta manera ens evitem la molesta carga de contingut que està allotjada en un servidor extern i que ens hem de baixar per a visualitzar. En comptes de tenir la informació fora i haver de connectar-se cada cop per a poder visualitzar-la és molt més senzill tenir totes les imatges, text, etc. dins de la aplicació i així poder accedir sense la necessitat d'Internet.

1.4 Objectius

El Treball Final de Grau té com a objectiu la creació d'una aplicació mòbil de venda d'una llista genèrica de productes que treballa a nivell offline excepte certes accions necessàries com la identificació d'un usuari, la realització d'una comanda, etc.

Aquesta aplicació estarà connectada a un servidor extern que atindrà totes les accions mencionades anteriorment.

A continuació es llisten els objectius que seguirà el TFG:

- Creació d'una aplicació mòbil d'un gestor de ventes de productes complint que:
 - Haurà de poder treballar a nivell offline en totes les accions excepte les totalment necessàries com pot ser la identificació d'usuari, el processament d'una comanda, etc.
 - Tingui una interfície agradable amb aparença semblant a una aplicació nativa.
 - Disposi de diferents tipus de vistes per a visualitzar de diferents maneres els productes (vista Simple, 3x3, 4x4 o en forma de llista).
 - Puguí ordenar de diferents maneres la llista de productes que es visualitzen.

- Disposi d'un buscador de productes per a poder localitzar o filtrar productes disponibles mitjançant els seus atributs.
- Un usuari haurà de ser capaç de crear o consultar llistes de productes preferides i comandes.
- Creació d'un servidor extern que sigui capaç de:
 - Tractar totes les peticions de la aplicació mòbil.
 - Tenir una base de dades de la informació dels usuaris, comandes, llistes i productes de la aplicació mòbil.

Conclusió: La aplicació mòbil ha de poder treballar a nivell offline per a evitar temps d'espera per a carregar dades d'Internet d'un servidor extern.

2. Anàlisi del tipus de desenvolupament

2.1 Elecció del tipus d'aplicació mòbil

Per a decidir quina tecnologia per al desenvolupament de la aplicació és la més òptima d'utilitzar, hem de saber quines diferents opcions existeixen per a desenvolupar una aplicació mòbil i extreure les avantatges i inconvenients que comporten cadascuna de les opcions.

Hi han varies opcions per al desenvolupament d'una aplicació mòbil:

- **Aplicació nativa:** S'utilitzarà el llenguatge del sistema operatiu del mòbil per a crear la aplicació. Per exemple, en el cas d'Android [*1] es programarà amb Java, Objective C per iOS [*2], etc. És la opció més viable si el que vols és treure tot el rendiment del mòbil. Pots accedir a tots els dispositius del mòbil com el GPS, agenda, acceleròmetre, etc. i fer que la aplicació funcioni amb una rapidesa immillorable. Les aplicacions són executades llavors pel propi sistema operatiu sense intervenir ningun dispositiu addicional. S'ha de crear un codi diferent per a cada un dels sistemes operatius per lo que comporta un preu de producció i manteniment més elevat.
- **Aplicació web:** Com el seu nom indica, les webs també es poden accedir des del mòbil a través del seu navegador web i poden resultar una aplicació profitosa i viable. S'ha de tenir en compte però, que la interfície de la web per a mòbil o smartphone no sol ser la mateixa que la de escriptori, i que per tant el punt a destacar és la creació de la interfície per a mòbil. El seu problema és que necessita d'Internet per a totes les accions ja que es basen en el model client-servidor, no pot accedir a molts dispositius del mòbil com poden ser la càmera, el GPS, l'acceleròmetre, etc. i la rapidesa de la aplicació dependrà d'Internet i del navegador del mòbil.

- **Aplicació híbrida:** Aquesta és la opció que ha aparegut recentment degut a la problemàtica creada pel fet que al coexistir diferents sistemes operatius comporta que si vols fer aplicacions natives hauràs de crear diferents codis per a la aplicació si vols arribar a tots els usuaris del mercat actual. Com que aquesta solució comporta un cost elevat, s'han creat diferents frameworks que combinen característiques de les dos anteriors. Per una part, obtindrem una aplicació que podrà accedir a tots els dispositius del mòbil, que podrà treballar offline igual que la aplicació nativa, i el seu únic codi basat en tecnologia web serà apte per a la majoria dels sistemes operatius. S'executarà mitjançant el navegador però es comportarà com si fos una aplicació nativa. El seu problema és que com que s'executa a través del navegador mai podrà ser igual de ràpida que una aplicació nativa però en molts casos la aplicació desitjada no comporta grans càlculs intensius i la rapidesa continuaria sent bona per a satisfer les necessitats del client si triéssim la opció de la aplicació híbrida.

En la Taula 1 es detallen les diferents característiques relacionades amb els objectius del projecte segons diferents tipus d'aplicacions.

	Processament	Offline	Multi Plataforma	Rapidesa Creació	Market Place	Accés Dispositius
Nativa	SO	Si	No	Alt per a MP	Si	Si
Web	Navegador	No X	Si	Baix	No	No
Híbrida	Navegador	Si	Si	Baix	Molts	Si

Taula 1: Comparació entre desenvolupament, natiu, web i híbrid.

En relació al projecte de realitzar una aplicació per a mòbil que pugui treballar online, s'han arribat a les següents conclusions:

La aplicació web queda descartada, ja que no pot treballar a nivell offline principalment. Tampoc podem gaudir de les avantatges de la distribució de la aplicació mitjançant els markets places de cada sistema operatiu, ni tampoc

podrem utilitzar eines com el GPS, accés als contactes per a publicitat, etc.

La següent decisió està relacionada en la tria d'aplicació nativa o híbrida.

2.2 Aplicació Nativa o Híbrida?

Un estudi realitzat per Kendo UI basat en la informació proporcionada per més de 5.000 desenvolupadors y responsables de tecnologia de diferents parts del món, que curiosament es titula 'El debate entre HTML5 y Nativo está acabado y el ganador es ...' [*3] han estat examinant tal decisió i sembla no haver-hi una resposta clara malgrat tot.

Tot i això, si que afirmen que encara que no existeixi una solució única que serveixi per tot, en els escenaris actuals hi ha una tendència clara a que els desenvolupaments Web i híbrids siguin cada cop més populars, escollint la combinació HTML5 + JavaScript com la millor solució per a projectes multiplataforma.

Hi han dos grans punts a tenir en compte per a poder decidir entre triar la opció nativa o la híbrida, el cost i temps de producció i la rapidesa de processament.

	Cost i Temps Desenvolupament	Rapidesa processament
Natiu	Alt	Excel·lent
Híbrid	Baix	Bona

Taula 2: Comparació entre desenvolupament natiu i híbrid.

Les dos van indirectament relacionades ja que una comporta l'altra.

Desenvolupament natiu: Si busquem que la nostra aplicació sigui molt ràpida inevitablement haurem d'escollir la opció del desenvolupament natiu.

Això ens comportarà que haurem de gastar-nos més recursos econòmics per al desenvolupament de cada sistema operatiu amb diferents llenguatges de

programació (Android-Java, IOS-Objective C, etc) ja que la opció nativa no és multiplataforma.

Per a grans empreses que busquin el millor producte i/o per a aplicacions que requereixin càlculs intensius com podrien ser molts dels jocs actuals la opció nativa és la adequada.

Desenvolupament híbrid: Si per altra banda, la aplicació que volem crear no requereix de grans càlculs intensius i la rapidesa de la execució de la aplicació no es veu compromesa, podem contemplar la opció del desenvolupament híbrid.

Ens estalviarem un gran cost de producció ja que obtenim un producte apte per a la majoria dels sistemes operatius mòbils existents fet amb un sol tipus de llenguatge i com a afegit podrem llençar al mercat el producte més ràpidament a tots els sistemes operatius ja que el temps de producció és més ràpid.

La majoria de frameworks híbrids solen utilitzar els llenguatges de JavaScript, Html i CSS que són llenguatges molt coneguts i utilitzats entre els programadors.

Existeixen previsions que apunten que al 2015, el 60% de les aplicacions mòbils seran híbrides, mentre que el percentatge de les basades en web i natives seran del 30% i 10% respectivament. [*4]

En el nostre cas, interessa crear una aplicació de gestió de comandes i no requereix de gran càlculs intensius, i per tant no es veurà compromesa la rapidesa de la aplicació en ningun dels casos.

L'elecció es basarà en el pressupost disponible que tinguem. Com la idea és arribar a tots els públics a un preu (temps o diners depenent del cas) que no sigui molt elevat s'escollirà la opció de desenvolupament híbrid.

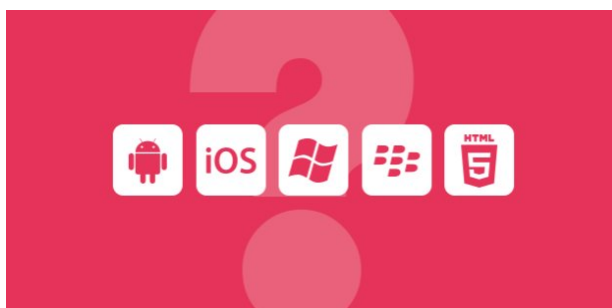
El fet de estalviar-se la contractació d'un informàtic expert en tots els llenguatges nadius de mòbils o contractar diferents informàtics per a obtenir un producte apte per a tots els sistemes operatius és un punt a destacar molt important.

Per la mateixa regla de tres, el manteniment també serà igualment fàcil de realitzar ja que només ho hauràs de fer en un únic codi.

L'aplicació haurà de complir que sigui una aplicació híbrida i per tant que incorpori:

- Capacitat de treball offline.
- Únic codi per als diferents SO basat en llenguatges basats en programació web que farà que sigui una aplicació multiplataforma.
- Accés als dispositius del mòbil com poden ser el acceleròmetre, la càmera, els contactes, xarxa, emmagatzematge, etc.
- Com a afegit tindrem accés als Market Place dels SO mòbils, amb els processos corresponents de aprovació i publicació que podrem utilitzar per a promocionar o vendre productes.

2.3 Investigació de Frameworks Híbrid



En el mercat actual, els frameworks híbrids disponibles d'avui en dia en són varis però només els 2 o 3 més importants dominen el mercat. Es poden trobar frameworks que es puguin desenvolupar via Web o via Windows/Mac/Linux, gratuïts o de pagament, enfocat per a aplicacions gràfiques o més senzilles en quan a processament, etc. que et donen tot un ventall de opcions segons les necessitats.

A continuació val la pena descriure breument els frameworks més coneguts:

- AppInventor [*5]: Encara que no és multiplataforma i només serveix per Android val la pena mencionar aquest framework web de la empresa Google que no necessita de coneixements de programació per a construir una aplicació senzilla. Mitjançant l'ús de blocs amb l'estil d'un trencaclosques, permet construir aplicacions que no requereixen d'un funcionament extensiu de llibreria Android. Un dels seus defectes és la necessitat d'Internet per a poder construir quelcom.

- Batuin [*6]: Crea de forma fàcil amb Java i Html aplicacions senzilles a través de la web i distribueix-les a través de les Market Places, QRCode, Wifi, SMS o Bluetooth.
- Corona [*7]: Pensat per al desenvolupament de jocs i aplicacions gràfiques per a Iphone/Android.
- Marmalade [*8]: Utilitzat per a la conversió de videojocs a plataformes mòbils mitjançant C++, Lua, Objective-C i HTML . Suporta les següents plataformes: iOS, Android, Bada, Symbian, WebOS.
- Phonegap [*9]: El més utilitzat i conegut utilitzat per més de 400.000 desenvolupadors. Està basat en els llenguatges web de HTML i CSS per a la interfície gràfica i amb JavaScript per a la gestió d'esdeveniments i accions. Es poden utilitzar llibreries addicionals com jQuery o Sencha per a un disseny que aparenti una aplicació mòbil nativa. No recomanada per a aplicacions intensives com jocs o aplicacions amb efectes visuals. Suporta la exportació als SO Android, iOS, WP7, BlackBerry, Symbian, Bada i WebOS. La propietària de Phonegap és la coneguda empresa Adobe. Phonegap ha guanyat el premi "People's Choice" en el 2009 a la Conferència Web 2.0. de San Francisco [*10].
- RhoMobile [*11]: S'utilitza el llenguatge Ruby per a la lògica de la aplicació i Html, CSS i JS per a la interfície i al igual que Phonegap exporta als mateixos SO excepte Bada. La empresa propietària d'aquest framework és Motorola que recentment ha estat comprada per Google.
- Titanium [*12]: Està basat íntegrament en JavaScript i el disseny resulta complex. El programador ha de construir-se ell mateix les seves pròpies interfícies amb components nadius com botons propis del mòbil. No obstant això, aquesta característica comporta un augment de la fluïdesa de les interfícies. Suporta la exportació als SO Android, iOS, BlackBerry, Tizen, Denso i Mobile Web. En el 2012, va guanyar el premi a la "Millor tecnologia en núvol", entre molts altres premis [*13].
- Trigger.io [*14]: Semblant a Phonegap però només és compatible per a iOS i Android. Utilitza el llenguatge web de Html, CSS i JavaScript. Amb Trigger.io es poden crear interfícies on la velocitat de resposta resulta molt més àgil ja que usa components nadius pròpies com la barra superior o la

barra de pestanyes. La compilació de les aplicacions al núvol resulta més ràpida que la de PhoneGap Build.

La següent taula es mostren les principals característiques de cadascun.

	<u>Preu</u>	<u>MP</u>	<u>Llenguatges</u>	<u>Plataforma Desenvolupament</u>	<u>Problemes</u>
ApplicationCraft	14 \$/mes	Android / iOS / WP*1	JavaScript	Web Online X	Disseny limitat, desenvolupament online
AppInventor	Si	Android X	Ningun	Web Online X	Només Android, web online
Batuin	495 o 865 € / any	Si	JavaScript i Html nivell baix	Web Online X	Apps senzilles
Corona	199 o 349 \$/any*2	Android/ IOs X	Lua	Windows Mac	Pensat per a fer Jocs X
Marmalade	149 o 499 \$/any *3	Si*3	C++ Lua Objective-C Html	Windows Mac	Poc conegut, Linux no disponible
Phonegap	Si*4	Si	Html JavaScript CSS	Windows Mac Linux	càlculs intensius
RhoMobile	Si	Si menys Bada	Html JavaScript CSS Ruby	Windows Mac Linux	Comunitat poc activa. No exporta a Bada
Titanium	Si*5	Android/ iOS X	JS XML	Windows Mac Linux	Disseny costós, documentació poc completa
Trigger.io	+59 \$/any	Android/ iOS X	Html CSS JavaScript	Windows Mac Linux	Documentació i suport poc complet

Taula 3: Frameworks de desenvolupament híbrids.

*1: Utilitza de tecnologia Phonegap per a la exportació als sistemes operatius.

*2: Corona és gratis pel desenvolupament però proporciona la opció de penjar les aplicacions a les app stores per un preu depenent del número de projectes actius.

*3: Marmalade proporciona només dues plataformes per el preu de 149\$ i si pagues 349\$ amplien les plataformes a Android, iOS, BlackBerry, Windows Phone

8 i Tizen

*4: Phonegap només deixa tenir un projecte actiu al núvol del Phonegap Build en la versió gratuïta.

*5: Titanium en la versió gratuïta no inclou uns complements que ofereix la empresa on realment les funcions interessants a utilitzar estan allà.

De la taula de característiques es pot veure que els frameworks ApplicationCraft, Batain i AppInventor són frameworks híbrids que treballen a nivell web on els desenvolupadors necessiten d'Internet per a poder treballar. Pel projecte que es planteja en aquesta memòria, no és una bona solució donat que el venedor o usuari és possible que no sempre disposi d'Internet. Independentment de l'ús d'Internet, el seu suport, documentació i serveis que proporcionen no semblen ser tant complets com altres frameworks que s'instal·len al PC com podria ser Phonegap, Titanium, etc.

Sobre Corona, es troba l'inconvenient de que és de pagament i està enfocat per al desenvolupament de jocs o aplicacions gràfiques.

En el cas de Marmalade, podria ser molt bona opció, però per a poder exportar a tots els sistemes operatius es necessita pagar el servei més car de 499 \$/any. Un altre punt negatiu és el poc coneixement d'alguns llenguatges com Lua i tampoc es pot treballar en Linux.

Analitzant Trigger.io es troben molts punts bons a destacar. És més ràpid compilar comparat amb el conegut PhoneGap Build de PhoneGap i es proporcionen uns mòduls, com el "native topbar" que es capaç de crear i controlar barres de control natives incloent botons i estils. El servei mínim proporcionat ofereix el poder treballar amb un sol projecte i un sol desenvolupador treballant i costa 39 \$/mes. La gran pega i la causa de no escollir aquest framework serà la única compatibilitat amb Android/iOS/Web.

Dins del conjunt d'entorns multiplataforma que han aparegut en el mercat, PhoneGap, Titanium i RhoMobile són les més conegudes i millor comentades en les

diferents referències existents, a més que tenen un gran nombre d'aplicacions en l'App Store i l'Android Market . Com a exemple de la importància d'aquestes tres plataformes podem citar el llibre "Pro Smartphone Cross – Platform Development" [*15], el qual es centra en aquests tres frameworks.

D'altra banda, aquestes tres plataformes també han rebut diferents premis. Titanium va rebre el premi 2010 Jolt Productivity Award de Dr Dobbs [*13], encarregat de premiar productes que han sacsejat el mercat i que es caracteritzen per fer de la creació de programari una mica més ràpid , senzill i eficient . RhoMobile va obtenir el premi Best Startup Company a l' edició 2009 Best of Interop Awards [*16], per ser considerada la millor entre les noves companyies que es van presentar a la convenció . Per la seva banda, PhoneGap va guanyar al Web 2.0 Expo LaunchPad a l'abril de 2009 [*10] , que també premia la millor entre les noves companyies .

En el següent apartat s'analitzen aquests tres frameworks híbrids més importants.

2.4 Frameworks Híbrids més importants

2.4.1 Phonegap



PhoneGap és un framework per al desenvolupament d'aplicacions mòbils produït per Nitobi, i comprat posteriorment per la empresa Adobe Systems (Offermann & Rios, 2011), fet que dóna una seguretat i estabilitat al producte ja que Adobe és una empresa internacional de molt renom.

PhoneGap és una solució de codi obert per a la creació d'aplicacions mòbils multiplataforma amb tecnologies web basades en estàndards com HTML, JavaScript i CSS3.

El servei de compilació de PhoneGap Build és gratuït però només proporciona la opció de un projecte actiu a la vegada per a exportar a qualsevol sistema operatiu. Si es desitja treballar amb molts projectes haurem de optar per a la opció de pagament de 10 \$/mes.

PhoneGap utilitza el navegador per executar l'aplicació i dóna la possibilitat de suportar funcions sobre altres frameworks com Sencha Touch [*17] o jQuery Mobile [*18], encara que poden repercutir negativament en la velocitat de execució perdent fluïdesa.

Exporta a tots els sistemes operatius mòbils existents més importants (Android, iOS, BlackBerry, Windows, Symbian i Bada) a partir d'un sol codi base, encara que no és del tot cert, ja que necessiten algunes petites modificacions.

Per a publicar l'aplicació a l'AppStore / Android Market amb un preu per descàrrega (aplicació de pagament) no es pot utilitzar PhoneGap perquè podrà ser "piratejada" amb extrema facilitat, en incloure el codi font.

La depuració d'aplicacions en PhoneGap és molt millor que els de Titanium, ja que depenen del Webkit estàndard que es pot depurar utilitzant eines de desenvolupament web.

Hi ha multitud de formes d'utilitzar PhoneGap per al desenvolupament d'aplicacions. Ha estat integrat en les últimes versions de Dreamweaver ©. Està disponible en format de plugin per a diferents programes, com el cas d'Eclipse [*19] o com template per Xcode [*20] en Mac.

2.4.2 RhoMobile

Tot i que les aplicacions desenvolupades mitjançant Rhodes s'instal·len i s'executen com aplicacions natives, el procés és molt similar al seguit per al desenvolupament web, seguint l'estàndard MVC (Model-View-Controller) [*21].

Rhodes és un gran framework open source per a la creació de forma ràpida d'aplicacions natives mitjançant el llenguatge Ruby per a la lògica (Model-Controller) i utilitza HTML, CSS i JavaScript per a la interfície (View).

Rhodes dóna la possibilitat d'utilitzar el maquinari integrat en el dispositiu (GPS, càmera, Contactes, NFC, Bluetooth etc.) de manera fàcil i senzilla a través d'unes API's que proporciona Rhodes.

Admet exportació i suport per a la majoria de sistemes operatius dels smartphone actuals. En concret iOS, Android, BlackBerry, Windows Mobile, Windows Phone 7 i Symbian. Bada per això encara no està incorporat.

RhoMobile Suite està integrada per RhoElements, Rhodes, RhoConnect, RhoStudio, RhoHub i RhoGallery.

Un cop desenvolupada l'aplicació, a partir del codi Ruby es genera el codi natiu de la plataforma que hagi estat seleccionada. Això es fa mitjançant RhoHub, que és la part de RhoMobile encarregada de la generació de les aplicacions.

2.4.3 Titanium



Titanium és una plataforma per al desenvolupament de aplicacions mòbils, per a tablets i per a escriptori utilitzant tecnologia web. Ha estat creada per la empresa Appcelerator Inc. al 2008. Té 1,5 milions de desenvolupadors i més de 20.000 aplicacions. La quantitat de documentació que hi ha per iniciar-se en aquest llenguatge és bastant gran. Dins d'aquesta gran comunitat els caps més visibles són companyies com eBay, PayPal que utilitzen aquesta plataforma per crear les seves pròpies aplicacions.

S'utilitza íntegrament JavaScript per al seu codi, creant i col·locant "a mà" tots els

controls, usant per a això una llibreria que fa de pont entre la teva aplicació JavaScript i els controls del sistema. Això significa que les finestres i altres controls visuals (botons, llistes, menús, etc) són nadius: quan s'afegeix un botó, es crea un botó del sistema i s'afegeix a la vista, fet que el fa més ràpid de renderitzar i la resposta de l'usuari és també la més ràpida.

En empaquetar l'aplicació, el JavaScript és transformat i compilat. Després, quan s'arrenca l'aplicació al mòbil, el codi s'executa dins d'un engine JavaScript, tal com diu la documentació oficial, que serà JavaScriptCore en iOS (l'interpret de Webkit, el motor de Safari i Chrome) i Mozilla Rhino en Android / BlackBerry.

El fet que el JavaScript estigui compilat i que els controls creats siguin nadius, li fa tenir el millor rendiment possible dins de les opcions de frameworks híbrids.

A diferència de PhoneGap, en Appcelerator no hi ha DOM [*22], de manera que no es poden usar llibreries com jQuery que serveixen per manipular, ja que el context d'execució és JavaScript pur, no dins d'un document HTML.

Per a programar es proporciona Titanium Studio, un IDE [*23] basat en Eclipse amb el qual crear els projectes i editar els fitxers JavaScript i la resta de recursos i llançar els scripts de creació. Experimentalment, permet usar Php, Ruby i Python també, però transformant el codi després en JavaScript amb els frameworks Phpjs, Skulpt o Ruby.js.

Titanium tradueix JavaScript a codi natiu en temps d'execució utilitzant un traductor incorporat en el paquet per a cada plataforma per lo que obtenim un resultat semblant al caràcter natiu.

Fins fa poc Titanium només exportava per a iOS, Android i Mobile Web. Recentment per això s'ha ampliat a BlackBerry i Windows.

2.5 Elecció Final del Framework Híbrid

	<u>A favor</u>	<u>En contra</u>
Phoneygap	Màxima exportació i llenguatges comuns	Evitar càlculs intensius o aplicacions gràfiques
RhoMobile	Molt complet, alta exportació	No exporta a Bada i la comunitat és menys activa respecte Phoneygap
Titanium	Aspecte natiu	No pot exportar a tots els SO, disseny costós

Taula 4: Comparació entre Phoneygap, RhoMobile i Titanium.

Titanium malgrat ser molt bona opció s'ha descartat degut a la poca compatibilitat en la exportació a sistemes operatius. Tampoc ajuda la poca facilitat del disseny de les interfícies.

RhoMobile té com a inconvenients que no té compatibilitat amb Bada i que la comunitat de desenvolupadors en Rhomobile no sembla tan extensa com la de Phoneygap.

Així doncs, la opció escollida serà PhoneGap ja que proporciona una exportació a la majoria de sistemes operatius mòbils, el llenguatge de desenvolupament es basa en els llenguatges webs més coneguts (HTML, CSS i JavaScript) i les funcionalitats que es necessiten per a la aplicació les pot efectuar Phoneygap sense cap impediment.

3. Anàlisi de l'aplicació

3.1 Anàlisi de compradors interessats

Els perfils de compradors d'aquest producte són les petites i mitjanes empreses que estiguin interessats en promocionar els seus productes o gestionar-los. Aquest projecte compleix aquesta premissa ja que es basa en un gestor de comandes de productes i molt fàcilment podem fer que aquesta aplicació sigui informativa per a fer publicitat únicament.

Les grans empreses haurien de quedar descartades ja que el producte que haurien d'escollir seria de desenvolupament natiu que encara que resulti més car i complex la aplicació resultant és més eficient.

3.2 Anàlisi d'usuaris

L'aplicació comptarà amb tres perfils d'usuari:

- Usuari no identificat: Només pot consultar el catàleg de productes i preparar la llista de la compra.
- Usuari identificat: Compta amb les accions del client no registrat més amb les accions de finalitzar la compra, repetir comandes anteriors i comandes preferides de l'usuari.
- Venedor o Usuari empresa: Compta amb les accions del client registrat però en comptes de comprar la comanda el mateix usuari, apareixerà l'usuari venedor com a supervisor de la comanda i afegirà les dades de l'usuari que es quedarà amb la comanda. És per aquesta raó que tindrà accés a la dades dels usuaris. Està pensat per a treballadors de la empresa per a promocionar les ventes mitjançant l'aplicació.

3.3 Anàlisi funcional

A continuació mostrarem totes les possibles pantalles que poden aparèixer a la aplicació mòbil.

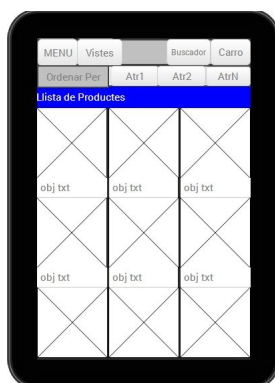
Primerament es mostren totes les vistes d'un usuari no identificat que són totes aquelles a les que es pot accedir sense necessitat d'Internet. Es poden consultar tots els productes i preparar la comanda al carro de la compra,

A les vistes de l'usuari identificat ja es poden consultar dades antigues del client a través d'Internet i per tant es poden mostrar les vistes de últimes comandes o comandes preferides de l'usuari.

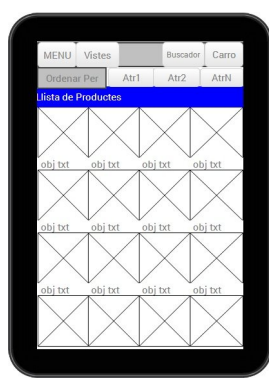
Les vistes del venedor seran les mateixes que les del client més les associades a les funcions d'aquest usuari com pot ser tramitar comandes i la consulta de informació de clients, etc.

3.3.1 Tipus de Vistes

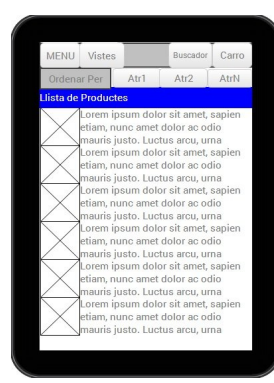
3.3.1.1 Vistes d'usuari no identificat



(a) Vista 3x3



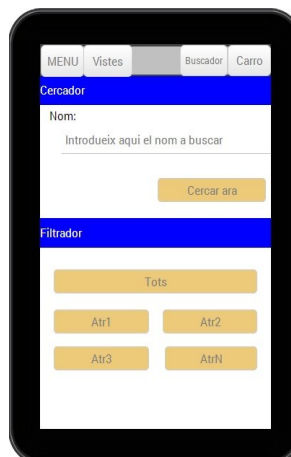
(b) Vista 4x4



(c) Vista Llista

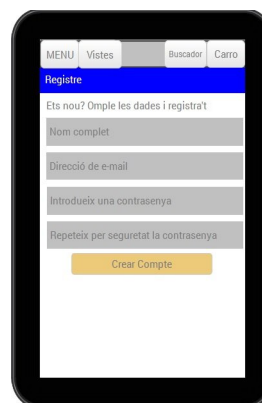
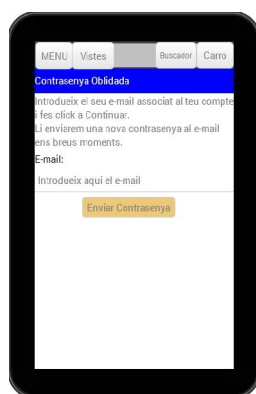
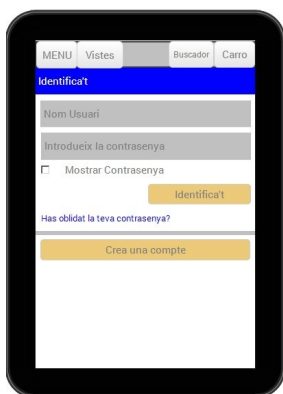
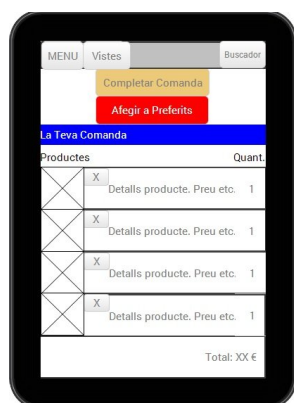


(d) Vista Simple



(e) Barra Superior

(f) Cercador



(g) Carro Compra

(h) Identificar
Usuari

(i) Recordar
Contrasenya

(j) Registrar
Usuari

Figura 1: Vistes usuari no identificat

A continuació es descriuen les vistes que s'usaran a l'aplicació per a un usuari no identificat i que es mostren a la Figura 1.

Vistes 3x3, 4x4, Llista (a)(b)(c): S'ofereixen 3 tipus de vista alternatives a la simple on es poden veure varis productes a la vegada de manera atractiva. Té una zona reservada per a ordenar els productes i prement qualsevol imatge o descripció es dirigirà a la vista Simple del productes que hem pitjat.

Vista Simple (d): És la vista encarregada d'afegir els productes al carro de la compra i d'ampliar la imatge i detalls. Disposa de dos botons per indicar la quantitat d'unitats d'un producte que desitgem afegir i un botó que afegirà aquest número en qüestió a la comanda.

Barra Superior (e): Disposa del botó Menú que enllaça amb la zona d'usuari. Disposa del botó Canvi de Vista que al dóna accés a tots els tipus de vistes (Simple, 3x3, 4x4, Llista). També disposa del botó d'una lupa per a poder cercar o filtrar la llista de productes, i el botó del Carro que t'enllaça amb la pantalla que gestiona la comanda personal actual.

Cercador (f): Es pot introduir un nom en concret per a trobar els productes relacionats amb ell i també hi ha una zona reservada per a filtrar la llista de productes a partir dels valors d'un atribut. En cas de trobar un sol producte et portarà a la Vista Simple.

Carro de la Compra (g): Es mostrarà una pantalla amb la llista dels productes seleccionats anteriorment especificant la quantitat de productes que s'ha triat de cada article i el total del preu que suma. Si es prem a la imatge s'amplia el producte en qüestió. Si es prem a la X eliminem el producte de la llista. També es pot modificar la quantitat final de productes. Si es vol afegir més productes es poden tornar a les vistes anteriors prement el botó de endarrere de la barra superior especial del carro de la compra.

Si la llista és correcte es pitjarà el botó de Completar Comanda. En cas que no s'estigui registrat saltarà la vista d'Identificar Usuari.

També podem guardar la llista de productes del carro com a llista preferida pitjant el botó de Agregar a Llistes Preferides.

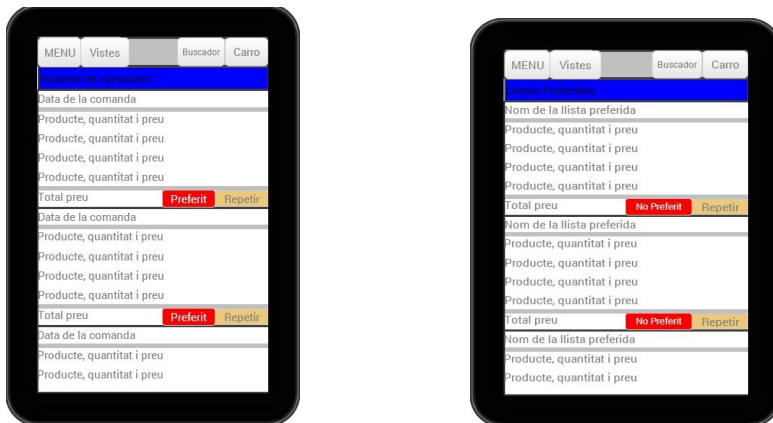
Identificar Usuari (h): Es Pot accedir a la pantalla d'identificació d'usuari a través del menú o al prémer Completar Comanda si l'usuari no s'ha identifica't o diferents situacions que requereixin de la identificació de l'usuari. Hi apareixen els enllaços per si no es recorda la contrasenya i/o per si no es disposa d'un compte creat.

Recordar Contrasenya (i): Conté un zona per introduir el correu electrònic associat a la compte d'on volem recordar per a rebre la contrasenya.

Creació Compte (j): A través de varis zones s'introdueix la informació necessària per a la creació d'un usuari registrat.

3.3.1.2 Vistes d'usuari identificat

Les vistes a les quals pot accedir un usuari identificat són totes les figures de l'usuari no identificat més les següents es mostren a la Figura 1 que s'accedeixen a través del botó Menú de la barra superior quan s'està identificat:



(a) Historial Comandes

(b) Llistes Preferides

Figura 2: Vistes usuari identificat.

A continuació es descriuen les vistes que s'usaran a l'aplicació per a un usuari identificat i que es mostren a la Figura 2.

Historial Comandes (a): Conté una llista de les últimes comandes realitzades per l'usuari identificat. En pitjar el botó Repetir d'una de les comandes s'accedirà al carro de la compra amb els productes de la llista en qüestió. També podem prémer el botó de Preferit per a guardar la llista de productes de la comanda desitjada com a una llista preferida introduint un nom a la llista que es demanarà.

Llistes Preferides (b): Conté una llista de les llistes de productes preferides de l'usuari identificat. En pitjar el botó Repetir en una de les llistes s'accedirà al carro de la compra amb els productes de la llista en qüestió. També es pot pitjar el botó No Preferit per a eliminar la llista preferida de la teva les seves llistes preferides.

3.3.1.3 Vistes de venedor

Les vistes a les quals pot accedir un venedor són totes les figures de l'usuari identificat modificant la de l'historial de comandes més la següent que es mostra a la Figura 3 (b) que s'accedeix al realitzar una comanda:



Figura 3: Vistes de venedor.

A continuació es descriuen les vistes que s'usaran a l'aplicació per a un venedor i que es mostren a la Figura 3.

Historial Comandes (a): Igual que en el usuari identificat però s'afegeixen dos botons a la zona inferior de la comanda que permetrà processar o cancel·lar el procés de la comanda en els estats "demanada" o "en procés". L'historial de comandes es subdivideix en Comandes Demanades, En procés, Cancel·lades o Completades per a una millor gestió de las comandes.

Llista Usuaris (b): Apareix al intentar realitzar una comanda i al pitjar un dels usuaris se'l adjudica la comanda a aquest usuari en concret.

4. Casos d'Ús

4.1 Registrar Usuari

Cas d'ús: Registrar Usuari
Objectiu: Permet registrar un nou usuari i accedir a les vistes d'usuari registrat
Actor: Usuari no identificat
Precondicions: -
Passos: <ol style="list-style-type: none">1. L'usuari accedeix al menú principal a través del botó de la esquerra de la barra superior.2. El sistema desplega les diferents opcions del menú.3. L'usuari selecciona la opció Registrar Usuari.4. El sistema mostra pantalla de registre d'usuari.5. L'usuari omple les dades requerides i pitja el botó Crear Compte.6. El sistema retornarà un missatge de validació.
Extensions: <ol style="list-style-type: none">5a. L'usuari no omple bé el format de les dades a introduir i pitja el botó dedicat a Crear Compte.<ol style="list-style-type: none">1. El sistema comunica a l'usuari que no ha introduït bé les dades.2. Es retorna al pas 5.6a. El sistema informa que les dades no són vàlides per a registrar-se o ja estan en ús.<ol style="list-style-type: none">1. Es retorna al pas 5.

4.2 Identificar Usuari

Cas d'ús: Identificar Usuari
Objectiu: Permet identificar un usuari i accedir a les vistes d'usuari registrat
Actor: Usuari no identificat
Precondicions: Haver realitzat en el passat el cas d'ús 5.1
Passos: <ol style="list-style-type: none"> 1. L'usuari accedeix al menú principal a través del botó de la esquerra de la barra superior. 2. El sistema desplega les diferents opcions del menú. 3. L'usuari selecciona la opció Identifica't. 4. El sistema mostra pantalla d'identificació d'usuari. 5. L'usuari omple les dades requerides i pitja el botó Identifica't 6. El sistema retornarà un missatge de validació.
Extensions: <ol style="list-style-type: none"> 5a. L'usuari omple unes dades incorrectes i pitja el botó Identifica't. <ol style="list-style-type: none"> 1. El sistema informa que no s'han introduït correctament les dades. 2. Es retorna al pas 4.

4.3 Recordar Contrasenya

Cas d'ús: Recordar Contrasenya
Objectiu: Enviar un e-mail recordatori de la contrasenya del compte de l'usuari.
Actor: Usuari no registrat, usuari registrat o venedor
Precondicions: Haver realitzat en el passat el cas d'ús 5.1
<ol style="list-style-type: none"> 1. L'usuari accedeix al menú principal a través del botó de la esquerra de la barra superior. 2. El sistema desplega les diferents opcions del menú. 3. L'usuari selecciona la opció Identifica't. 4. El sistema mostra pantalla d'identificació d'usuari. 5. L'usuari pitja l'enllaç "Ha oblidat la seva contrasenya?". 6. El sistema mostra la secció de "Ha oblidat la seva contrasenya?". 7. L'usuari introdueix el email associat al compte i prem el botó de "Enviar Contrasenya".

8. El sistema retorna un missatge de confirmació.

Extensions:

8a. El sistema informa que les dades no són les correctes o s'ha produït un error al servidor.

1. Es retorna al pas 6.

4.4 Localitzar Producte

Cas d'ús: Localitzar Producte

Objectiu: Consultar la llista de productes actual a través de les diferents vistes per localitzar un producte interessant i visualitzar finalment el producte en forma de vista Simple.

Actor: Usuari no registrat, usuari registrat o venedor

Precondicions: -

Passos:

1. L'usuari pitja el botó en forma d'ull de la barra superior.
2. El sistema desplega una barra secundària amb les 4 vistes disponibles a escollir.
3. L'usuari escull una vista (3x3, 4x4 i Llista).
4. El sistema mostra la llista de productes en la vista seleccionada.
5. L'usuari selecciona un producte dins la llista.
6. Els sistema mostra en forma de vista Simple el producte desitjat.

Extensions:

3a. L'usuari escull la vista Simple.

1. El sistema mostra el primer producte de la llista o últim producte visitat.
2. L'usuari cerca el producte desitjat mitjançant les fletxes de navegació.
3. Es retorna al pas 6.

4.5 Ordenar Llista de Productes Actual

Cas d'ús: Ordenar Llista de Productes Actual

Objectiu: Ordenar la llista de productes actual de l'usuari segons un dels diferents atributs preparats per a poder ordenar-se.

Actor: Usuari no identificat, usuari identificat o venedor

Precondicions: Estar en la vista 3x3, la vista 4x4 o la vista Llista

Passos:

1. L'usuari pitja l'atribut de la zona reservada al principi de la llista pel qual vol ordenar la llista actual.
2. El sistema ordena de manera descendent segons l'atribut.

Extensions:

- 2a. El sistema ordena de manera ascendent si l'últim cop que s'havia activat s'havia ordenat de manera descendent segons l'atribut.

4.6 Cercar Per Nom

Cas d'ús: Cercar Per Nom

Objectiu: Utilitzar el cercador implementat per a trobar productes del catàleg sencer a partir d'un nom.

Actor: Usuari no identificat, usuari identificat o venedor

Precondicions: -

Passos:

1. L'usuari pitja el botó de la barra superior en forma de lupa.
2. El sistema mostra la pantalla del cercador i filtrador.
3. L'usuari introdueix un nom a cercar i pitjar el botó de "Cercar ara".
4. El sistema visualitza els productes que continguin aquest nom en forma de vista Llista i el botó inicial en forma de lupa canvia de color per indicar que la llista de productes actual no és la total.

Extensions:

- 4a. El sistema informa que no ha trobat ningun resultat.
 1. L'usuari prem el botó "D'acord".
 2. Es retorna al pas 2.
- 4b. El sistema només troba un resultat en la cerca i es mostra forma de vista Simple i el botó inicial en forma de lupa canvia de color per indicar que la llista de productes actual no és la total.

4.7 Filtrar Productes

Cas d'ús: Filtrar Productes

Objectiu: Filtrar o reduir la llista de diferents maneres per a una fàcil localització

de productes interessants.
Actor: Usuari no identificat, usuari identificat o venedor
Precondicions: -
<p>Passos:</p> <ol style="list-style-type: none"> 1. L'usuari pitja el botó de la barra superior en forma de lupa. 2. El sistema mostra la pantalla del Cercador i Filtrador. 3. L'usuari prem el botó del nom de l'atribut per al qual vol filtrar. 4. El sistema mostra tots els valors possibles d'aquest atribut. 5. L'usuari selecciona el valor del que està interessat. 6. El sistema visualitza en forma de vista Llista els productes que continguin aquest valor d'atribut i el botó inicial en forma de lupa canvia de color per indicar que la llista de productes actual no és la total.
<p>Extensions:</p> <ol style="list-style-type: none"> 5a. L'usuari selecciona un valor que només conté un resultat. <ol style="list-style-type: none"> 1. El sistema visualitza en forma de vista Simple el producte que conté aquest valor d'atribut i el botó inicial en forma de lupa canvia de color per indicar que la llista de productes actual no és la total.

4.8 Afegir Producte al Carro de la Compra

Cas d'ús: Afegir producte al carro de la compra
Objectiu: Afegir producte al carro de la compra de la comanda
Actor: Usuari no identificat, usuari identificat o venedor.
Precondicions: venir del cas d'ús 5.4.
<p>Passos:</p> <ol style="list-style-type: none"> 1. L'usuari selecciona la quantitat del producte i prem el botó d'afegir. 2. El sistema mostra un missatge de que s'ha afegit correctament el producte al Carro de la Compra.
<p>Extensions:</p> <ol style="list-style-type: none"> 1a. L'usuari no introdueix una quantitat vàlida i prem el botó d'afegir. <ol style="list-style-type: none"> 1. El sistema mostra un missatge d'error. 2. Es va al cas d'ús 5.8.

4.9 Preparar comanda

Cas d'ús: Preparar una Comanda
Objectiu: Realitzar una comanda de productes
Actor: Usuari no identificat, usuari identificat o venedor.
Precondicions: haver realitzat el cas d'ús 5.8 mínim 1 cop.
<p>Passos:</p> <ol style="list-style-type: none"> 1. L'usuari pitja el botó de la barra superior dedicat al carro de la compra. 2. El sistema mostra la pantalla del Carro de la Compra amb tots els productes afegits. 3. L'usuari modifica la quantitat o elimina algun dels productes afegits i es torna al pas 2.
<p>Extensions:</p> <ol style="list-style-type: none"> 3a. L'usuari no modifica la quantitat ni elimina algun dels productes afegits.

4.10 Realitzar una comanda.

Cas d'ús: Realitzar una Comanda
Objectiu: Realitzar una comanda de productes
Actor: Usuari identificat o venedor
Precondicions: venir del cas d'ús 5.9
<p>Passos:</p> <ol style="list-style-type: none"> 1. L'usuari Pitjar el botó de Completar Comanda de la pantalla del carro de la compra. 2. El sistema pregunta si està segur de completar la comanda. 3. L'usuari prem que si. 4. El sistema retorna un missatge de confirmació.
<p>Extensions:</p> <ol style="list-style-type: none"> 2a. El sistema detecta que l'usuari no està identificat. <ol style="list-style-type: none"> 1. L'usuari realitza el cas d'ús 5.2. a partir del pas 4. 2. Es retorna al pas 2. 2b. El sistema detecta l'usuari utilitza l'aplicació de venedor i apareix una

pantalla on demanarà afegir el client al qui es ven la comanda.

1. El venedor prem el client a qui processar la comanda.

2. Es retorna al pas 2.

3a. L'usuari prem que no.

1. Es retorna al cas d'ús 5.10.

4.11 Localitzar l'historial de comandes

Cas d'ús: Localitzar l'historial de comandes
Objectiu: Localitzar una comanda passada realitzada anteriorment
Actor: Usuari identificat o venedor
Precondicions: Estar registrat e identificat (cas d'ús 5.1 i 5.2).
Passos: <ol style="list-style-type: none">1. L'usuari prem el botó Menú de la barra superior.2. El sistema mostra les diferents opcions del menú.3. L'usuari selecciona la opció d'Historial de Comandes.4. El sistema mostra la pantalla de comandes amb la informació de les últimes comandes realitzades.
Extensions: <ol style="list-style-type: none">3a. El venedor selecciona una de les opcions de les Comandes (Demandedes, En Procés, Completades o Cancel·lades).<ol style="list-style-type: none">1. Es retorna al pas 4.4a. El sistema informa que no té comandes realitzades.

4.12 Crear Llista Preferida del Carro de la Compra

Cas d'ús: Crear Llista Preferida del Carro de la Compra
Objectiu: Crear una llista de productes preferida per l'usuari a partir de la llista de productes del Carro de la Compra.
Actor: Usuari identificat o venedor
Precondicions: venir del cas d'ús 5.9 o venir del cas d'ús 5. i haver realitzat en el passat el cas d'ús 5.10

Passos:

1. L'usuari prem el botó d'Afegir a Llistes Preferides
2. El sistema mostra una pantalla per a la confirmació de la creació de la llista preferida.
3. L'usuari prem que Si.
4. El sistema informa que s'ha creat correctament la llista preferida.

Extensions:

- 3a. L'usuari prem que No.
 1. Es retorna al cas d'ús 5.12
- 4a. El sistema informa que no s'ha pogut crear correctament la llista preferida.
 1. Es retorna al cas d'ús 5.12

4.13 Localitzar Llista Preferida

Cas d'ús: Localitzar Llista Preferides

Objectiu: Localitzar una llista de productes preferida de l'usuari.

Actor: Usuari identificat, Venedor

Precondicions: Haver realitzat en la sessió actual el cas d'ús 5.2.

Passos:

1. L'usuari prem el botó Menú de la barra superior.
2. El sistema mostra les diferents opcions del menú.
3. L'usuari selecciona la opció de Llistes Preferides.
4. El sistema mostra les llistes preferides de l'usuari.

Extensions:

- 4a. El sistema informa que no té llistes preferides.

4.14 Elaborar el Carro de la compra a partir dels productes d'una Comanda o Llista Preferida

Cas d'ús: Elaborar el Carro de la compra a partir dels productes d'una Llista Preferida

Objectiu: Els productes de la llista preferida escollida passaran a ser els productes

del carro de la compra.
Actor: Usuari identificat, Venedor
Precondicions: Estar al pas 4 del cas d'ús 5.11 al pas 4 del cas d'ús 5.14.
<p>Passos:</p> <ol style="list-style-type: none"> 1. L'usuari prem el botó d'afegir productes al carro de la compra. 2. El sistema mostrarà una pantalla de confirmació de l'actualització del carro de la compra. 3. L'usuari prem la opció Si. 4. El sistema mostra la pantalla del carro de la compra amb els nous productes.
<p>Extensions:</p> <ol style="list-style-type: none"> 3a. L'usuari prem la opció No. <ol style="list-style-type: none"> 1. El sistema mostra el pas 4 del cas d'us 5.11 o 5.14 respectivament.

4.15 Eliminar llista preferida

Cas d'ús: Eliminar una llista preferida de l'usuari
Objectiu: S'eliminarà la llista preferida de l'usuari.
Actor: Usuari identificat o venedor
Precondicions: Estar al pas 4 del cas d'ús 5.11 o al pas 4 del cas d'ús 5.14.
<p>Passos:</p> <ol style="list-style-type: none"> 5. L'usuari prem el botó d'afegir productes al carro de la compra. 6. El sistema mostrarà una pantalla de confirmació de l'actualització del carro de la compra. 7. L'usuari prem la opció Si. 8. El sistema mostra la pantalla del carro de la compra amb els nous productes.
<p>Extensions:</p> <ol style="list-style-type: none"> 3a. L'usuari prem la opció No. <ol style="list-style-type: none"> 1. El sistema mostra el pas 4 del cas d'us 5.11 o 5.14 respectivament.

4.16 Canviar d'estat una comanda.

Cas d'ús: Canviar d'estat una comanda
Objectiu: El venedor podrà executar els canvis de l'estat de la comanda dels usuaris.
Actor: Venedor
Precondicions: estar al pas 4 del cas d'ús 5.11 Localitzar Historial de Comandes o al pas 4 del cas d'ús 5.14 Localitzar Llista Preferida.
Passos: <ol style="list-style-type: none">1. L'usuari prem el botó de processar la comanda.2. El sistema mostrarà una pantalla de confirmació del processament de la comanda.3. L'usuari prem la opció Si.4. El sistema mostra la pantalla de confirmació de canvi d'estat.
Extensions: <ol style="list-style-type: none">3a. L'usuari prem la opció No.<ol style="list-style-type: none">1. El sistema mostra el pas 4 del cas d'us 5.11 o 5.14 respectivament.

5. Disseny

5.1 Explicació del disseny

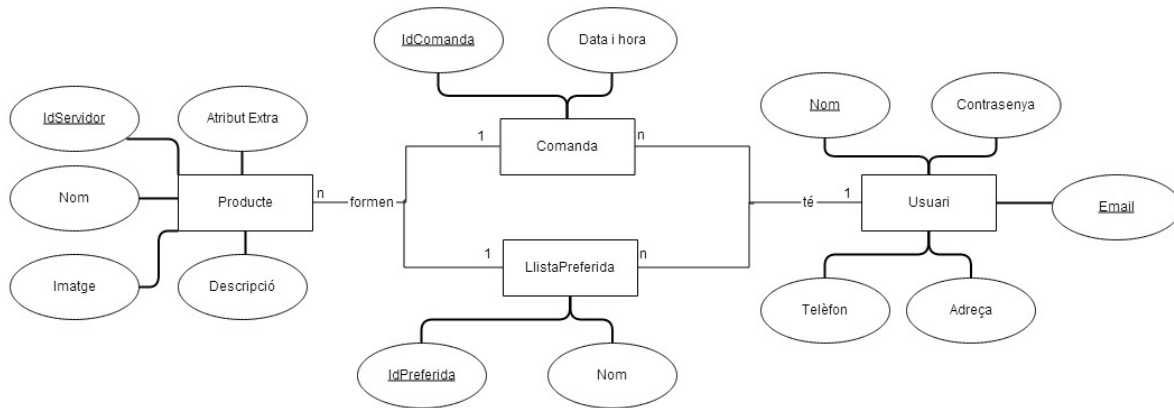


Figura 4: Diagrama de classes

La estructura de la base de dades de l'aplicació segueix la mostrada per la Figura 4 on molts productes poden formar una comanda o una llista preferida. Al seu temps, moltes comandes o llistes preferides poden ser associades a un usuari. Així doncs, un usuari podrà tenir moltes comandes o llistes preferides que podran contenir molts productes.

Per a buscar una solució al temps de resposta de la aplicació s'ha basat l'aplicació en el que es mostra a la Figura 5. En aquesta figura es mostra l'arquitectura funcional de l'aplicació a desenvolupar, detallant cadascuna de les parts i tecnologies utilitzades a cada component. Bàsicament, és una aplicació client-servidor, on el client allotja l'aplicació local i en el servidor resideix el model de negoci i el control dels usuaris.

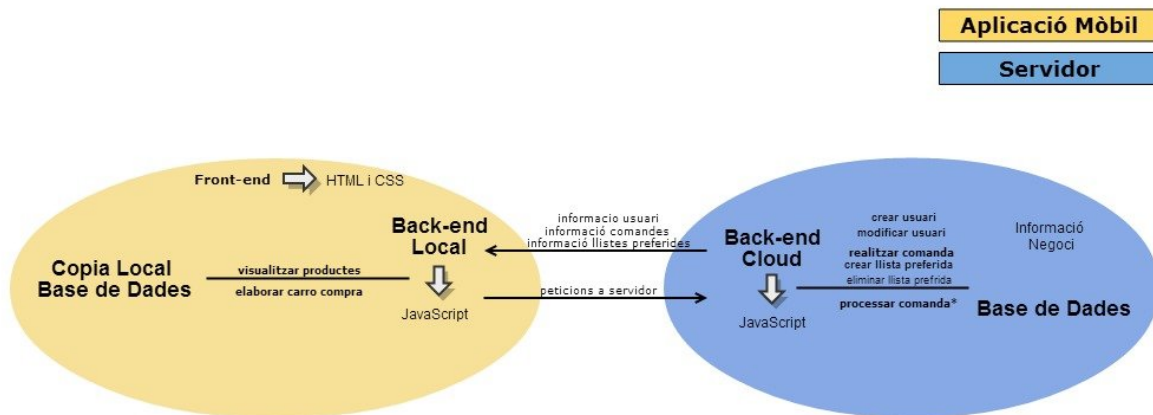


Figura 5: Funcionament de l'aplicació

L'aplicació local contindrà tot el codi HTML, CSS i JavaScript necessari per a la realització de les tasques no relacionades amb un usuari identificat com pot ser la realització d'una comanda o la consulta o modificació d'informació personal que el servidor haurà de gestionar. També disposarà d'una còpia de tota la informació que es pugui copiar de la base de dades sense posar en perill la seguretat dels usuaris. En aquest cas, s'ha fet una còpia de tota la informació dels productes del servidor a l'aplicació necessària per al bon funcionament i dóna la possibilitat d'alliberar d'informació no necessària al servidor.

D'aquesta manera es pot assolir l'objectiu de poder treballar offline sense necessitat d'estar demanant constantment la informació dels productes utilitzant el llenguatge HTML i CSS per l'aspecte de l'aplicació i JavaScript per al com interactua l'aplicació.

Per a la interfície, s'utilitza el sistema DOM que utilitza la tecnologia web separant dues seccions principals, la barra superior o header i el contingut que està dins del cos o body del html que s'utilitza i s'aplica un disseny mitjançant css per a la majoria de accions encara que algun cop s'utilitza javascript d'ajuda per a modificar algun valor relacionat amb l'aspecte d'una capsula o div.

Per resoldre el problema de les diferents resolucions de pantalla dels dispositius mòbils s'ha creat un css per a cadascun dels intervals de resolucions més lògics extrets de l'estudi de les resolucions més utilitzades pels dispositius mòbils en la actualitat.

D'aquest estudi s'han extret 5 arxius de css per a cadascuna d'elles però és fàcilment ampliable per a perfeccionar els dissenys resultants.

El servidor utilitzat en aquest TFG és un servidor web molt conegut anomenat Parse.com on ofereix una base de dades orientada en objectes estable i un servei de cloud computing o servei de núvol on es podran gestionar les peticions dels usuaris de manera còmoda i senzilla amb el llenguatge ja conegut en la aplicació mòbil de javascript i unes funcions proporcionades per Parse.com per a poder modificar i accedir a la informació de la base de dades. S'ha de tenir en compte que les peticions a funcions del servidor són asíncrones però disposa certes funcions atòmiques per a un bon ús de sincronisme.

S'ha d'evitar utilitzar càlculs intensius ja que Phonegap crea una aplicació que s'executa mitjançant el navegador i no directament en el sistema operatiu del mòbil pel que és més lent que una aplicació nativa. Per a solucionar aquest problema s'ha tingut en compte diferents opcions de millora de processament:

- Phonegap té compatibilitat amb llibreries com jQuery Mobile o SenchaTouch però es recomana no utilitzar-les si es vol aconseguir la màxima velocitat de processament en la aplicació ja que aquestes solen alentir els efectes i processament de l'aplicació i en aquest cas s'evitava tot tipus de alentiment.
- Un altre mètode molt eficient que s'ha utilitzat en aquest TFG per a guanyar velocitat en el processament és només carregar un cop un html i que en comptes d'anar movent-se d'un html a l'altre es modifiqui només el contingut del propi html obert simulant que s'està canviant de html quan el que s'està fent és romandre al mateix html sempre canviant el contingut quan es precis.

- En la secció de motivació del TFG ja s'ha mencionat aquesta tècnica però no està relacionada directament amb els càlculs intensius sinó amb el temps de càrrega d'informació del servidor. I és que, si evitem demanar informació al servidor i podem obtenir la informació de la mateixa aplicació evitem un temps d'espera valuós que farà creure a l'usuari que la aplicació és ràpida en el processament. Només en accions d'usuari com la identificació, modificació de informació personal i tractament de comandes o llistes preferides necessàries en el servidor s'utilitzarà l'accés a Internet.

5.2 Millores extretes de simulacions i proves

Entre altres aspectes que s'han canviat del disseny inicial pensat és pot destacar la utilització d'una mida de lletra més gran de la esperada, ressaltar les paraules importants en negreta, l'ús variat de colors suaus i la localització de certs botons en la interfície ha estat essencial sobretot per a la gent d'edat més avançada.

Es va intentar inicialment fer una doble barra d'opcions, una superior com la que s'utilitza finalment i una inferior. La barra inferior resultava molt poc atractiva per aplicacions que es permeti girar la pantalla en horitzontal i que s'utilitzi el teclat per a introduir dades ja que l'espai restant queda molt reduït i poc atractiu. Hauria estat bona opció per a aplicacions informatives que no necessitin la utilització de teclat o que no permetin el gir en horitzontal.

S'han realitzat diferents test d'usabilitat amb 3 informàtics graduats i 10 possibles usuaris, 3 entre 40 i 60 anys, 3 de menys de 18 anys i 4 entre 18 i 40 anys d'edat per a anar millorant la interfície inicial pensada fins a la actual interfície utilitzada per a una millor realització dels casos d'ús d'un usuari. Malgrat els petits canvis l'usuari ha pogut realitzar les tasques amb més o menys rapidesa però tots han coincidit que és agradable la velocitat d'interactuar amb la aplicació.

S'han realitzat diferents proves dels casos d'ús que s'expliquen a continuació amb imatges d'ajuda. Entre elles, escollirem els casos d'ús d'identificar usuari i el cas

d'ús de realitzar comanda amb les precondicions incloses. Els usuaris no han tingut cap ajut per a la realització de les proves i s'expliquen les conclusions extretes de cadascuna en els següents apartats.

5.2.1 Prova cas d'ús "Identificar d'usuari"

A continuació s'explica el flux normal que un usuari hauria de seguir sense cap impediment i es mostra a la Figura 6 unes imatges aclaridores:



Es prem el botó del número 1 de la Figura 6 i apareix el secció inferior just sota del botó pitjat. Es pitja el botó que indica el número 2 i apareix la pantalla de la segona imatge. S'omple els camps dels números 3 i 4 i es prem el botó Identifica't mostrat al número 5.

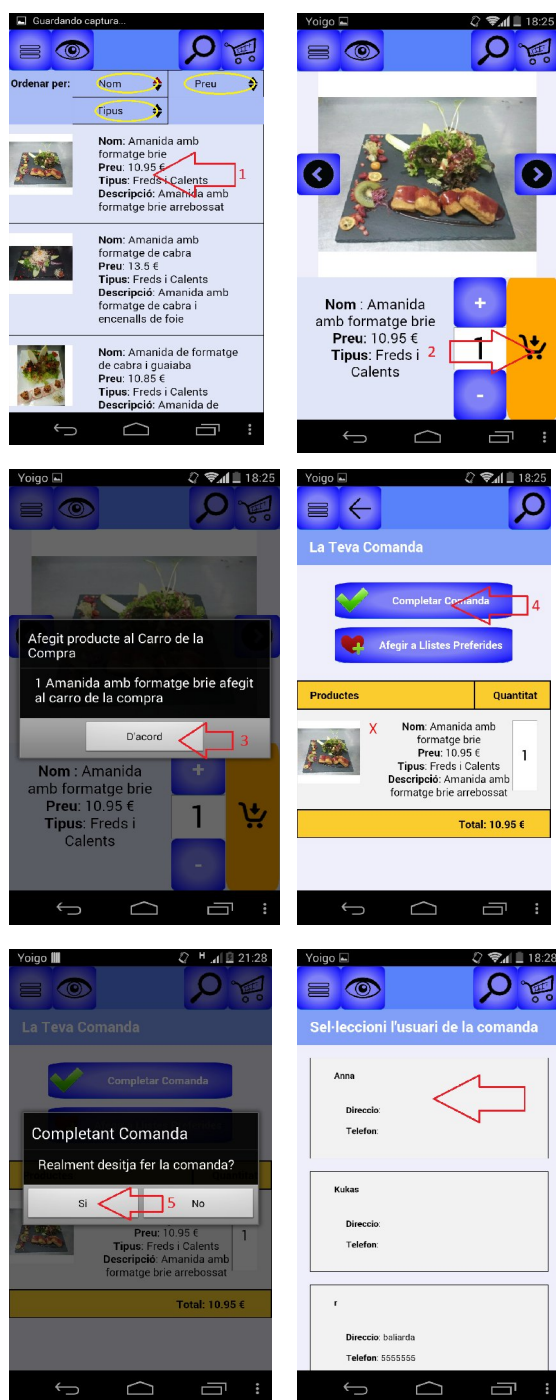
Si no hi ha cap error de servidor apareixerà un missatge del sistema que tot ha succeït com s'esperava.

Figura 6: Imatges aclaridores del cas d'ús "Identificar usuari"

El problema més important a destacar en aquest cas d'ús és la identificació del primer pas. La gent d'edat avançada tardaven més temps del que s'esperava i però acabaven trobant el botó del Menú. A partir d'aquí ja intuïen fàcilment el que s'havia de realitzar. Com a conseqüència d'aquest fet, s'ha utilitzat un icona on la majoria d'usuaris associés el concepte Menú amb millores detectables en la usabilitat.

5.2.2 Prova cas d'ús "Afegir producte, preparar i realitzar Comanda"

A continuació s'explica el flux normal que un usuari hauria de seguir sense cap impediment i es mostra a la Figura 7 unes imatges aclaridores:



En la primera imatge de la Figura 7 veiem sobresaltats amb cercles grocs la zona on es pot ordenar la llista de productes.

Al prémer la zona d'un dels productes com per exemple el 1 de la primera imatge se'ns mostra la vista Simple del producte escollit i al pitjar el botó indicat del número 2 l'afegim al carro.

El sistema informa quants productes d'aquell tipus s'ha afegit al carro de la compra. Es repeteix tants cops com productes vulguem al carro i al prémer el botó en forma de carro de la barra superior es veu la 4arta imatge.

Per a finalitzar es pitja el botó completar comanda.

Per seguretat es preguntarà per últim cop si està segur de realitzar la comanda.

La última imatge que es mostra que demana seleccionar un usuari pertany al usuari venedor.

Figura 7: Imatges aclaridores del cas d'ús "Afegir producte, preparar i realitzar Comanda"

El problema més important a destacar en aquest cas d'ús és la localització del botó del carro quan ja s'han afegit tots els productes. La gent d'edat avançada tardaven més temps del que s'esperava i però acabaven trobant el botó del Carro. A partir d'aquí ja intuïen fàcilment el que s'havia de realitzar. Com a conseqüència d'aquest fet, s'ha utilitzat un icona on la majoria d'usuaris associés el concepte de la zona de compra amb la imatge del carro amb millores detectables en la usabilitat.

Altre problemes indirectes d'aquest cas d'ús és la precipitació de prémer el botó del Carro abans d'hora però s'havien rectificar l'error tornant a la zona dels productes.

També succeïa que buscaven el botó d'Afegir al carro en les llistes plurals de productes però intuïtivament pitjaven un dels productes i entenien que el botó en qüestió estava en la vista Simple que apareix al prémer qualsevol producte.

Al afegir un producte al carro tornar a les llistes plurals a través del botó de les vistes no sempre era obvi i la majoria d'ells intuïtivament pitjaven el botó natiu d'endarrere fet que s'ha implementat així per evitar malentesos.

5.2.3 Proves de l'aplicació en diferents negocis

La aplicació resultant és versatilitat amb diferents bases de dades. Com a exemples principals s'han utilitzat dos temes de negoci: el restaurant Petit Bonet i una cerveseria imaginària.

Les aplicacions resultants funcionen adequadament i a continuació mostrem a la Figura 8 imatges del negoci de la cerveseria imaginària ja que el cas principal del restaurant Petit Bonet ja es mostren imatges aclaridores en altres apartats.



Figura 8: Mostra de l'aplicació en una altra llista de productes.

Veiem com s'adapta de forma correcta al nou negoci. Un exemple d'adaptació és el tenir un quart atribut pel que es pot ordenar. En el negoci del restaurant el atribut de la graduació corresponia a la descripció que no permetia el filtrat ni la ordenació. També veiem com existeix l'atribut d'origen que ocupa el lloc de l'atribut tipus del restaurant però a efectes pràctics, en aquest cas, funciona igual.

6. Conclusions i possibles ampliacions

6.1 Conclusions

En la actualitat, la fragmentació en el desenvolupament d'aplicacions per dispositius mòbils ha estat un problema molt present fins a l'actualitat. L'aparició i proliferació dels diferents sistemes operatius i les seves respectives botigues d'aplicacions, han fet que el desenvolupament tingui problemes afegits ja que si es vol tenir disponible l'aplicació per a les diferents plataformes s'haurà de realitzar més d'un codi de desenvolupament amb el cost afegit que correspon.

La importància de facilitar l'accés a les petites i mitjanes empreses i a particulars per al desenvolupament d'aplicacions mòbils que pugui proporcionar els serveis desitjats d'aquests subjectes a un preu més assequible de desenvolupament han propiciat a l'aparició de frameworks híbrids que permeten la creació d'aplicacions mòbil web que es comporten com a aplicacions natives i en la majoria de sistemes operatius.

El primer objectiu que s'ha plantejat en aquest TFG va ser triar entre desenvolupament web, híbrid i natiu. S'han analitzat les avantatges i desavantatges entre les tres opcions i s'ha arribat a la conclusió que la millor opció, en aquest cas i tal i com s'explica en el punt 3 del TFG, és la híbrida.

Concretament, s'ha realitzat un estudi sobre els diferents frameworks existents més coneguts, analitzant les principals característiques de cadascun. Finalment, i basant-nos amb les necessitats descrites del projecte, ens centrem amb les tres plataformes més interessants: PhoneGap, Titanium i Rhomobile.

Aquests tres frameworks són els que tenen més número de desenvolupadors, les més comentades i millor analitzades en les diferents referències existents, i les que posseeixen un major número d'aplicacions en la App Store i l'Android Market.

De l'anàlisi d'aquestes plataformes es pot afirmar que:

- Ninguna plataforma és superior a les demés. Cadascuna té els seus punts forts i els seus punts dèbils.
- Les diferents plataformes suporten la majoria de funcionalitats dels dispositius.
- La elecció de la plataforma vindrà determinada per les prioritats del desenvolupador i el tipus d'aplicació a crear en cada cas.

Finalment es tria Phonegap com a framework final degut a les següents raons:

- Compatibilitat amb la majoria de sistemes operatius.
- Compatibilitat amb la majoria de dispositius del mòbil.
- Comunitat de desenvolupadors molt activa i nombrosa.
- La compra del framework per la companyia Adobe demostra que és un framework molt viable en el futur.
- Els altres dos frameworks comportaven alguna característica no tan interessant o més costosa com la que oferia Phonegap.
- Hi ha mètodes beneficiosos en el disseny de l'aplicació per a aconseguir un rendiment òptim malgrat no ser un framework apta per a aplicacions que requereixin càlculs intensius.

Resumidament, Phonegap és un framework que es basa en els llenguatges de HTML, CSS i JavaScript, on els dos primers s'encarreguen del front-end o "com es veurà" la aplicació i la última s'encarrega del back-end o dit d'una manera senzilla el "que es veurà". Així doncs, és un framework de creació d'aplicacions mòbils molt interessant per a desenvolupadors ja experimentats en el món web.

Així doncs, l'aplicació creada en aquest TFG s'ha desenvolupat en base Phonegap i s'ha creat un manual tècnic en el annex que ajudarà a futurs desenvolupadors principiants a entendre l'entorn de desenvolupament de les aplicacions elaborades amb Phonegap. Tanmateix s'ha creat un apartat en al annex per als futurs usuaris que s'explica com instal·lar en un dispositiu mòbil l'aplicació de Phonegap.

El desenvolupament d'aplicacions amb altres entorns multiplataforma, ens permetria aprofundir en altres punts de vista dins del desenvolupament d'aplicacions mòbils i comparar la qualitat de les aplicacions resultants. Com a conseqüència, es tindria una visió més global de la funcionalitat dels diferents frameworks.

6.2 Possibles ampliacions o variacions

Els requisits de la aplicació han estat assolits però la aplicació té moltes possibles ampliacions o variacions que fàcilment es poden desenvolupar. A continuació en mencionem uns quants que tenen cert interès.

- La variació més lògica és convertir la aplicació en purament informativa i resultaria un producte excel·lent.
- Afegir el lliscament horitzontal o el zoom és possible i senzill.
- Millores de disseny amb efectes visuals.
- Funcions estadístiques de comandes i productes serien fàcils de desenvolupar en el servidor per a un millor servei al venedor.
- Utilització del GPS per a la millor indicació de les adreces dels usuaris o de la empresa.
- Creació d'una funció que permeti actualitzar la informació local de la aplicació a través del servidor.
- Millorar el buscador.

7. Referències

- [*1] Android: <http://www.android.com/meet-android/>
- [*2] iOS: <http://www.apple.com/es/ios/what-is/>
- [*3] "El debate entre HTML5 y Nativo está acabado y el ganador es ...":
<http://www.telerik.com/whitepapers/kendo-ui/the-html5-vs.-native-debate-is-over-the-winner-is>
- [*4] Previsió 2015 ús d'aplicacions híbrides:
<http://www.europapress.es/portaltic/software/noticia-aplicaciones-hibridas-representaran-60-apps-2015-20140109151118.html>
- [*5] AppInventor: <http://appinventor.mit.edu/explore/>
- [*6] Batuin: <http://batuin.aplicacionesmovil.com/marketing/es/home>
- [*7] Corona: <http://coronalabs.com/products/corona-sdk/>
- [*8] Marmalade: <https://www.madewithmarmalade.com/>
- [*9] Phonegap: <http://phonegap.com/>
- [*10] Premi Phonegap 2009: <http://www.oreilly.com/pub/pr/2270>
- [*11] RhoMobile: <http://www.motorolasolutions.com/US-EN/Business+Product+and+Services/Software+and+Applications/RhoMobile+Suite>
- [*12] Titanium: <http://www.appcelerator.com/titanium/>
- [*13] Premis Titanium: <http://www.appcelerator.com/company/awards/>
- [*14] Trigger.io: <https://trigger.io/>
- [*15] Pro Smartphone Cross – Platform Development: <http://it-ebooks.info/read/1945/>
- [*16] Premi RhoMobile: <http://www.prnewswire.com/news-releases/rhomobile-named-best-startup-in-the-2009-best-of-interop-awards-62006102.html>

- [*17] Sencha Touch: <https://www.sencha.com/products/touch/>
- [*18] jQuery Mobile: <http://jquerymobile.com/>
- [*19] Eclipse: <http://www.eclipse.org/>
- [*20] Xcode: <https://developer.apple.com/xcode/>
- [*21] MVC: <http://ca.wikipedia.org/wiki/Model-View-Controller>
- [*22] DOM: [http://ca.wikipedia.org/wiki/Model d'Objectes del Document](http://ca.wikipedia.org/wiki/Model_d'Objectes_del_Document)
- [*23] IDE : [http://ca.wikipedia.org/wiki/Entorn integrat de desenvolupament](http://ca.wikipedia.org/wiki/Entorn_integrat_de_desenvolupament)

8. Annex I: Manual Tècnic

8.1 Requeriments i instal·lació de la aplicació per a un usuari

La aplicació mòbil està optimitzada per al sistema operatiu mòbil Android però també es pot exportar a iOS, BlackBerry, Windows Mobile, etc. com s'explica a punts anteriors parlant de Phonegap. Pel cas d'Android és compatible des de la versió 4.0.0 amb una versió optimitzada.

Es disposa d'un codi Qr que permet la instal·lació de l'aplicació mòbil només escanejant el codi amb un lector de codi Qr.

També podem instal·lar la aplicació mòbil seguint els passos 8.2.5, 8.2.6 i 8.2.7.

També està disponible la pàgina web disponible des de qualsevol dispositiu que tingui d'un navegador.

8.2 Requeriments i instal·lació de l'entorn de desenvolupament del projecte

Aquest projecte Phonegap s'ha desenvolupat amb el sistema operatiu de 64-bits Windows 7 i s'han utilitzat els següents programes:

- Java SE, Apache Ant, Node.js, Phonegap, Eclipse i els SDK's d'Android.

Si es vol desenvolupar el projecte en base iOS es necessitaria un Mac.

8.2.1 Instal·lació de Node.js

És necessari per a poder instal·lar Phonegap. Podeu descarregar-vos la última versió de Node.js del següent enllaç:

<http://nodejs.org/download/>

La instal·lació no comporta cap possible imprevist. Només cal limitar-se a pitjar següent o acceptar segons la situació. És possible si teniu el Firewall de Windows activat us demani permís i caldrà acceptar.

8.2.2 Instal·lació Apache ant

Us podeu descarregar la última versió d'Apache (1.9.3 al Gener del 2014) de la següent adreça web:

<https://ant.apache.org/bindownload.cgi>

Extraieu la carpeta al lloc on voleu instal·lar Apache i redirigiu el path. S'aconsella canviar de nom la carpeta extreta per a un nom més senzill com "ant" i així també l'adreça que s'utilitza al redirigir el path és més còmode.

Per a redirigir el path ens situarem a la carpeta "Path" de la secció "Variables del Sistema" tal i com s'indica per arribar al punt 8.2 i pitjarem a "Editar". Al final de tot de la fila del "Valor de la variable" afegiu la ruta de la carpeta d'Apache, separant amb ";" de l'últim valor.

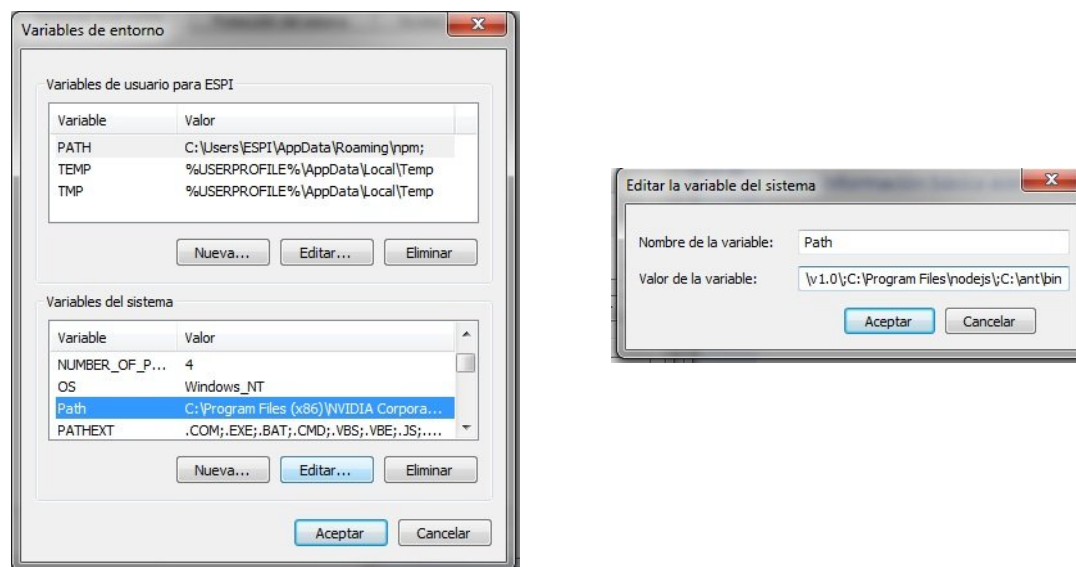


Figura 9: Configurant els paths d'Apache.

En acabat executeu "ant" per consola. Hauríeu de rebre un missatge com el següent:

```
Buildfile: build.xml does not exist!  
Build failed
```

8.2.3 Instal·lació de Java Standard Edition (Java SE)

Podeu descarregar-vos la última versió de Java SE (a Març del 2014 és la versió 7u51) del següent enllaç:

<http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>

També dóna la opció de baixar un pack junt amb Net Beans però només escollirem descarregar el Java Platform (JDK) 7u51.

Per a que l'entorn de Phonegap detecti adequadament Java SE s'han de redirigir el path de Java SE.

Primerament anirem al Escritori, pitjarem botó dret al icona de "Equipo" i seguidament "Propiedades".

Buscarem a la barra de la esquerra la opció de "Configuración avanzada del sistema" i seguidament a la part inferior pitjarem a "Variables de Entorno...".

A la secció de "Variables del Sistema" veuràs una pantalla com la següent:

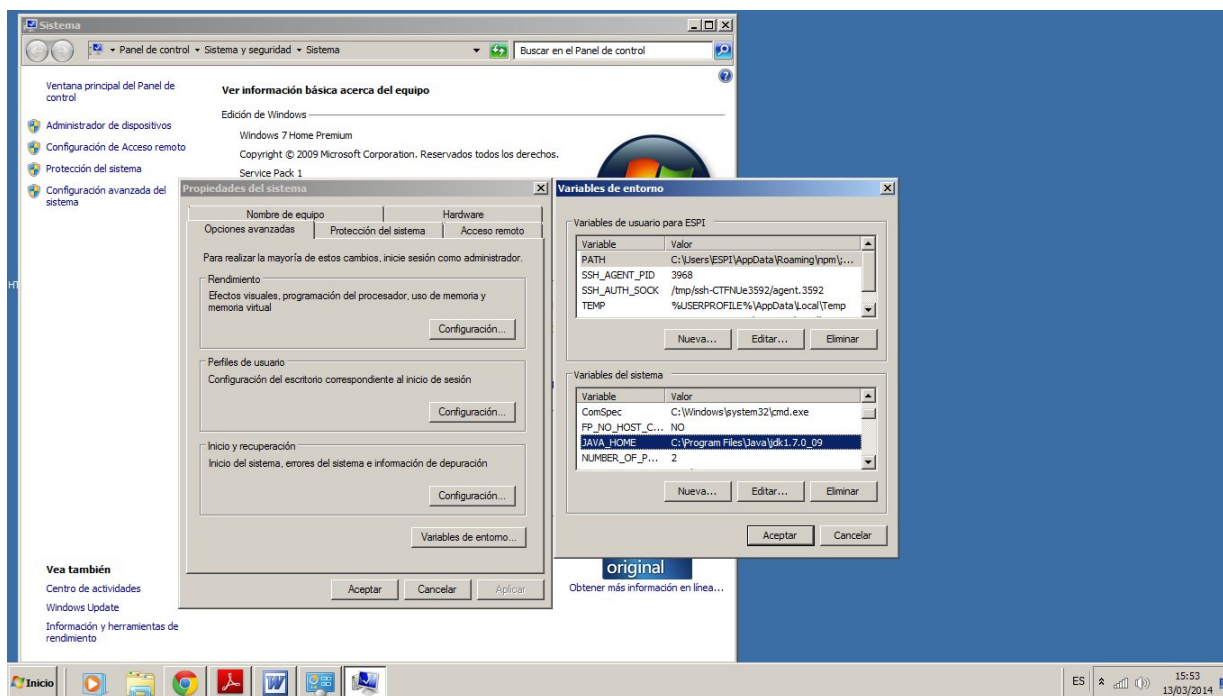


Figura 10: Configurant els paths Java

En aquesta imatge es veu com s'ha afegit la nova variable que detecta el path de Java SE pitjant a "Nueva..." on el nom de la variable del sistema és "JAVA_HOME" i s'ha indicat com a valor l'adreça d'on s'ha instal·lat el JDK. En el meu cas es situa a "C:\Program Files\Java\jdkNumVersio".

Seguidament a la variable "PATH", s'afegeix ";%JAVA_HOME%\bin" al final de tot de la fila del "Valor de la variable".

Podem comprovar que s'ha redirigit bé el path de Java SE podem executar per la cmd la comanda "Javac -version" i hauria de rebre per resposta "javac numVersió"

8.2.4 Instal·lació de Phonegap

Al de Març la versió que s'instal·la és la 3.4.0. Malgrat existir una versió més recent, Phonegap Build no suporta de moment exportar a BlackBerry, Symbian i WebOS per a una versió igual o superior de PhoneGap 3. Sabent això escollim la versió anterior possible, la 2.9.0 per a aquests sistemes operatius i la versió més recent per a Android i iOS. Només cal accedir a l'enllaç següent:

<http://phonegap.com/install/>

A la part inferior podem escollir la versió 2.9.0 d'on s'obté l'arxiu "phonegap-2.9.0.zip".

Per últim només caldrà extreure el contingut on voleu desar el programa.

8.2.5 Instal·lació d'Eclipse amb SDK's d'Android

Al mes de Març del 2014 hi ha disponible per a descarregar un pack que inclou una carpeta referent al IDE Eclipse, una carpeta pels SDK's d'Android, més un llançador anomenat "SDK Manager" per a poder descarregar els SDK's d'Android fàcilment.

Es pot localitzar a la següent adreça web:

<https://developer.android.com/sdk/index.html>

Seguidament caldrà descarregar totes les versions que necessiteu a través del SDK Manager. La següent imatge mostra com funciona.

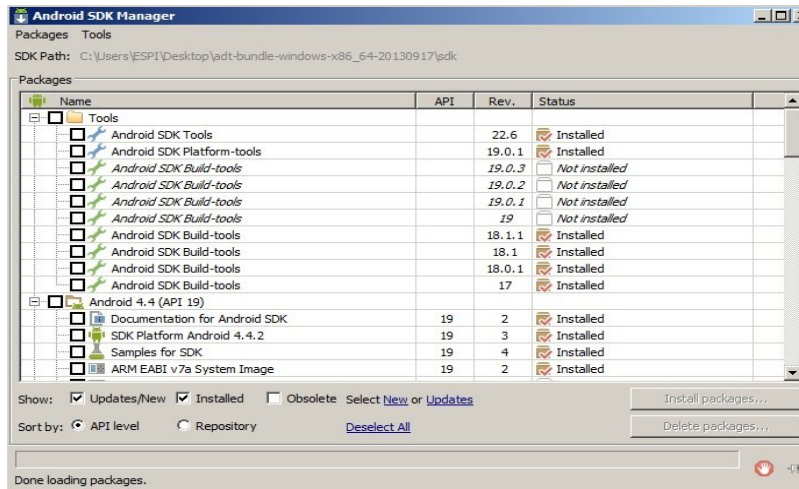


Figura 11: Instal·lant SDK's d'Android.

Un cop descarregats i actualitzats els SDK's, instal·lem el pluguin ADT Android necessari per poder desenvolupar aplicacions. Obrim Eclipse i a la barra superior premem l'opció Help > Install New Software. En la part que posa "Name" col·loquem ADT i en la part que posa "Location" afegim el següent i premem "Add":

"<https://dl-ssl.google.com/android/eclipse/>".

Ens apareixeran les eines a instal·lar, seleccionem les dues i li donem a Next. Posteriorment acceptem els termes i condicions de les eines que anem a instal·lar i començaran a instal·lar. Un cop fet això et sortirà una finestra que et demanarà reiniciar l'Eclipse, reiniciem i després reiniciar observaràs que apareixen nous icones a la barra d'eines, els quals ens indiquen que les eines s'han instal·lat correctament.

Com que el pack no s'instal·la sinó que és un executable caldrà redirigir els paths dels SDK's d'Android per a poder ser localitzat per a l'entorn de Phonegap. Es seguirà el mateix procediment que en el pas 8.3. i afegirem a Path aquestes dos adreces referents als SDK's.

"C:\Users\ESPI\Desktop\adt-bundle-windows-x86_64-20130917\sdk\platform-tools\; C:\Users\ESPI\Desktop\adt-bundle-windows-x86_64-20130917\sdk\tools\;"

8.2.6 Importació del projecte Phonegap

Per a poder exportar aquest projecte al IDE d'Eclipse seguirem els següents passos:

- Entrar al eclipse. Import->Android->Existing Android Code Into Workspace.
- Pitgem el botó "Browse..." i busquem la carpeta del projecte creat anteriorment i acceptem.

És possible que l'Eclipse t'utilitzi com a "Project Build Target", un SDK més antic que la versió 4 t'hagis descarregat en el projecte importat predeterminadament. En l'arxiu "AndroidManifest.xml" pot donar algun error ja que és incompatible amb versions antigues d'Android.

Per a corregir aquest error cal assegurar-se de canviar la target prement botó dret a la carpeta del projecte->Properties->Android-> i seleccionem el SDK més alt que tinguem instal·lat. Actualment la versió més recent és la "Android 4.4.2 Kit Kat".

Els arxius que utilitzarem per a exportar el projecte a Phonegap Build que s'explicarà al punt 8.8 són tots aquells que estiguin dins de la carpeta www dins de assets. Concretament podrem afegir arxius css, js i html com a principals tipus d'arxius però també podem afegir imatges, xml, etc.

Així doncs, dins de la carpeta assets/www/ estarà tot el cos del codi de la aplicació que es vol desenvolupar .

En la versió 3.4.0 Phonegap s'oculta la carpeta "www" i l'arxiu "config.xml" però es poden fer visibles prement botó dret a la carpeta del projecte->Properties->Resource->Resource Filters i eliminant els dos elements que hi apareixen.

Cal mencionar que existeix un pluggin per a Eclipse que et permet crear projectes Phonegap sense la necessitat d'instal·lar Apache, Node.js i Phonegap de la manera anterior però comporta el problema que la versió dels projectes Phonegap creats per aquest pluggin són de versions antigues i només arriba a la versió 1.9.0.

8.2.7 Testejar el projecte TFG Phonegap al IDE Eclipse

8.2.7.1 Simular mitjançant AVD

Una de les maneres de testejar el resultat del projecte és simular el projecte Phonegap a través del IDE Eclipse mitjançant el AVD (Android Virtual Device).

Podem localitzar la icona del AVD a la barra superior d'Eclipse, més a la dreta dels icones de crear i guardar. S'obrirà una pantalla en el que s'haurà de crear un

dispositiu Android virtual com el seu nom indica. La imatge a continuació mostra com crear-ho:

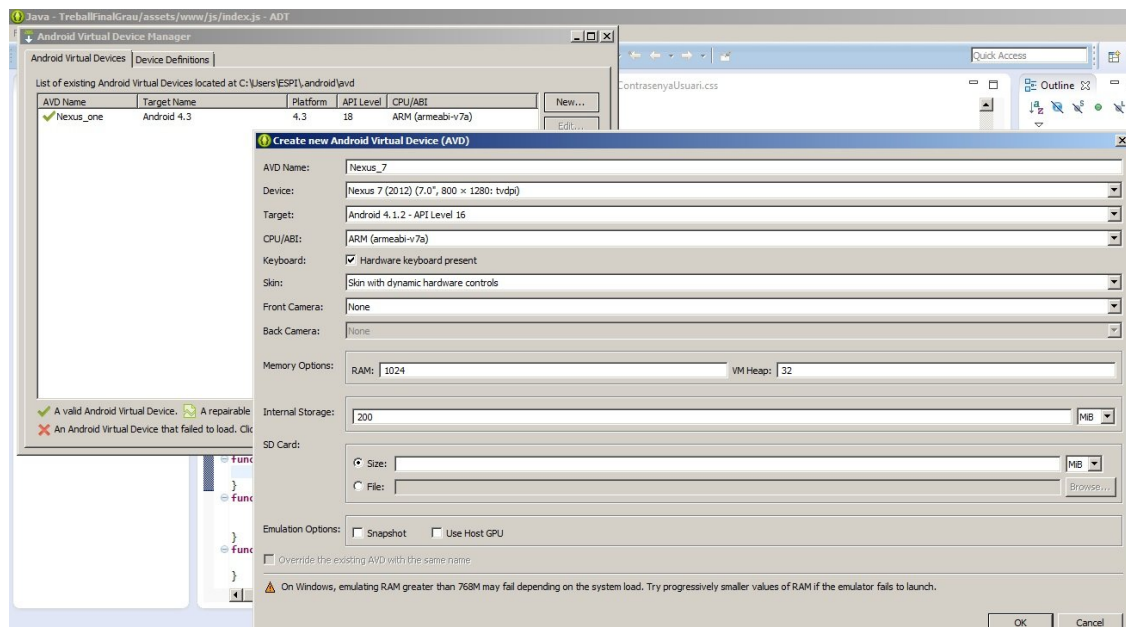


Figura 12: Creant un nou dispositiu Android virtual d'Eclipse

És possible que degut a un error creat per la nova actualització del ADT a Eclipse no funcioni correctament la creació d'un nou dispositiu al pitjar "OK" al final de la creació. Es pot accedir a la mateixa secció sense passar per Eclipse que permet la creació correctament anant a la carpeta del adt que inclou els SDK's i l'Eclipse. Dins de la carpeta "sdk", trobarem l'arxiu "AVD Manager.exe". Executeu l'arxiu i apareixerà la pantalla anterior i hauria de funcionar correctament.

Un cop instal·lat el dispositiu virtual, només cal pitjar el botó de "Run.." localitzada a la mateixa barra d'icones que es troben el New, Save, AVD, etc. El trobem a la part del mig.

8.2.7.2 Executar mitjançant un dispositiu mòbil Android

Podem connectar qualsevol dispositiu Android via USB al ordinador i apretant "Run As..." s'instal·larà i executarà el projecte Phonegap automàticament al vostre dispositiu.

Només cal tenir en compte els punts següents:

- Cal assegurar-se d'haver instal·lat prèviament els drivers del mòbil al ordinador sinó Eclipse no detectarà el mòbil correctament e intentarà llançar el AVD per a simular el projecte.
- Cal activar el mode "Depuración USB" del seu dispositiu mòbil Android. Per a activar-ho ho trobaràs a Ajustes->Opciones de desarrollo->Depuración USB normalment. No tots els dispositius s'activa la opció en qüestió amb el mateix procediment. Es pot consultar al manual del dispositiu en cas de dificultat o buscar per Google com activar el "modo programador" o el "modo desarrollo" del teu dispositiu smartphone o mòbil.
- Un altre punt a tenir en compte és comprovar la compatibilitat de la versió del sistema operatiu del projecte amb la del mòbil. El SDK mínim i màxim que acceptarà el projecte s'indica a l'arxiu "config.xml".

8.2.8 Compilar i Exportar amb Phonegap Build

Phonegap Build permet compilar i exportar el projecte creat en base Android a qualsevol dels altres sistemes operatius en que treballa Phonegap d'una manera molt senzilla i ràpida sense necessitat de SDK's.

Només cal registrar-se introduint les dades que es mostren a la imatge següent i la versió gratuïta et dona la opció de poder tenir un projecte actiu gratuïtament com es mostra a la part dreta.

← → ↻ https://build.phonegap.com/plans/free-adobeid

Bd Home Plugins Docs Blog FAQ Support Sign in Register

Register with an AdobeID

Create an Adobe ID

Adobe ID (Email Address)
jdoe@domain.com

Password

Retype Password

First Name

Last Name

Country/Region
Spain

Stay informed via email about Adobe products and services. [Learn more.](#)

Back Create

Free Plan [change](#)

1 private app
unlimited open-source apps
unlimited collaborators

Figura 13: Imatge de la pantalla de registre a PhoneGap Build

L'arxiu que pujarem a PhoneGap Build ha de ser un .zip que contingui la carpeta "www" que es situa dins del projecte a la següent direcció:

```
"...\platforms\android\assets\www"
```

És recomanable que l'arxiu config.xml estigui present dins d'aquesta carpeta ja que PhoneGap Build l'utilitzarà per a editar la informació de la aplicació com el títol de l'aplicació, icones, pantalles d'inici, i altres propietats.

Per a saber com configurar l'arxiu config.xml podem consultar el enllaç web següent:

http://docs.build.phonegap.com/en_US/3.1.0/configuring_basics.md.html

PhoneGap requereix un arxiu JavaScript diferent per a cada plataforma i utilitzant l'arxiu "phonegap.js" o "cordova.js" del projecte Android que hem desenvolupat resultarà incompatible per a les altres plataformes en executar l'aplicació.

És per això que eliminarem del .zip l'arxiu "phonegap.js" o "cordova.js" però la referència a l'arxiu .js en qüestió ha de continuar estant al codi de l'index.html.

Per a pujar a PhoneGap Build el projecte anirem a l'apartat de l'usuari Apps i pitjarem "Upload a .zip file".

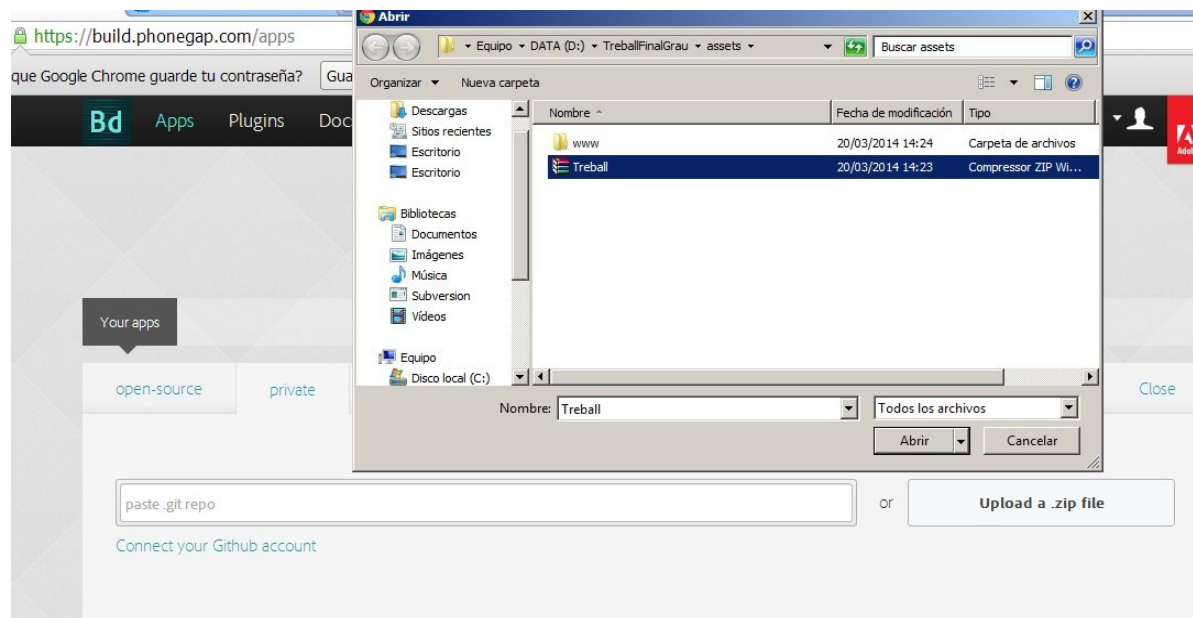


Figura 14: Selecció del projecte.zip per a PhoneGap Build

Seguidament carregarà el projecte i apareixerà una pantalla semblant a la següent.

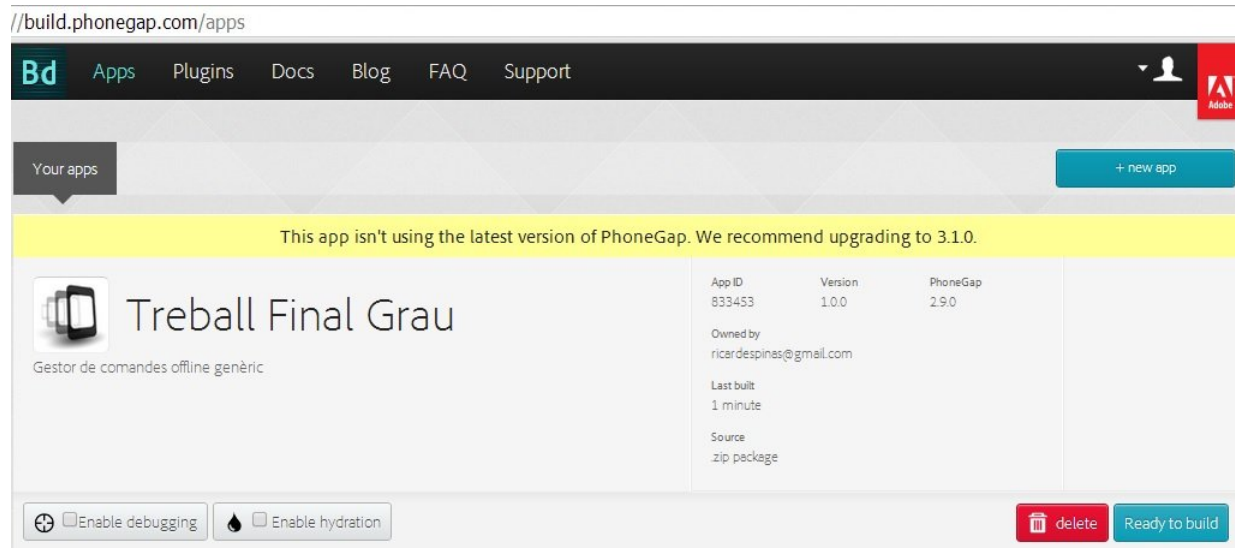


Figura 15: Pantalla d'informació del projecte que volem compilar

Pitjarem "Ready to build".

Si exportem per al desenvolupament per als sistemes operatius que no siguin Android o iOS amb la versió 2.9.0 apareixerà un missatge que recomana actualitzar la versió de PhoneGap que podem ignorar ja que hem escollit expressament aquesta versió ja que a partir de la versió 3.0.0 no permet exportar als altres sistemes operatius.

Si pel cas contrari, si que volem exportar per a Android o iOS, es necessita exportar a partir de la versió 3.0.0 controlant de forma afegida els permisos als dispositius que podrà accedir la aplicació mitjançant l'arxiu "config.xml".

<https://build.phonegap.com/apps/833453/builds>

Figura 16: Projecte pujat al PhoneGap Build disponible per a diversos sistemes operatius mòbils

En aquest punt, ja tenim la nostra aplicació disponibles per als diferents sistemes operatius mòbils que ofereix PhoneGap Build. L'error que apareix de iOS és resultat de la falta d'una "signing key" per a iOS.

Tenim els següents enllaços web per a saber com signar les vostres aplicacions per a iOS i Android.

iOS - http://docs.build.phonegap.com/en_US/3.1.0/signing_signing-ios.md.html#iOS%20Signing

Android - http://docs.build.phonegap.com/en_US/3.1.0/signing_signing-android.md.html#Android%20Signing

8.3 Requeriments i instal·lació del servidor del projecte

Parse.com no només té compatibilitat en JavaScript sinó que també en té amb iOS, Android, Windows Mobile, Windows Phone 8, Unity i OS X.

En el servidor hem de separar dos conceptes: base de dades i el núvol.

La base de dades funciona a través de la web, registrant-te i accedint a la secció personal on et deixa crear projectes amb uns serveis gratuïts bàsics i un servei més ampliat de pagament. L'elegit en aquest cas segons els requeriments de la aplicació serà el gratuït.

El núvol s'instal·la a Windows en aquest cas a través del programa Parse Tool on crea una carpeta amb diferents carpetes, entre elles la carpeta cloud, que conté l'arxiu .js on es podran definir les funcions que gestionaran les peticions dels usuaris.

També està disponible per a Linux. A continuació indiquem un enllaç de la guia d'instal·lació del núvol a Windows.

https://parse.com/apps/quickstart#cloud_code/windows